Step-Establishing Algorithm in Wireless TDMA Systems

LEE, King Ho

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Information Engineering

August 2008

Abstract of thesis entitled:

Step-Establishing Algorithm in Wireless TDMA Systems

Submitted by LEE, King Ho

for the degree of Master of Philosophy

at The Chinese University of Hong Kong in August 2008

In this thesis, we investigate the scheduling issues in the TDMA system. Our target is to develop a practical algorithm which utilizes scarce radio spectrum in the most efficient way, while satisfying the quality of service (QoS) requirement. In the first part of the thesis, we study various wireless scheduling models and power assignment schemes in the literature. It is shown that interference-based model with nonlinear power assignment scheme is the ideal approach for a protocol to find the optimal scheme of a network. In the second part of the thesis, we discuss two scheduling approaches, two-phase and step-removal, in the nonlinear power control scheduling scheme. After that, we provide a detailed analysis of two algorithms, Low-Disturbance Scheduling Algorithm (LDS) and Step-Removal Algorithm (SRA), each of which belongs to one of these two scheduling approaches. We show that both scheduling approaches have fundamental performance limitations on certain network topologies. In the last part of the thesis, we propose an algorithm called Step-Establish Algorithm (SEA), which is derived from SRA but has some subtle differences in that 1) SRA takes link-removal approach but SEA considers link-adding; 2) SRA uses normalized gain matrix Z in its checking stage but SEA employs matrix $\Phi = \beta^2 Z \circ Z^T$ in the checking process. From the proofs

and simulation results, it is found that SEA requires less time slots to schedule a set of transmission requests than SRA and LDS, resulting in the best performance. As the future work, we would like to investigate if there exists any network topologies which SEA requires more time slots than SRA and LDS to schedule given transmission links.

Acknowledgement

I would like to show my sincerest gratitude to my supervisors for giving me endless supports, insightful guidance, and constructive criticism, as well as their expertise on the subjects that arose throughout this thesis. Their enthusiasm and dedication to their students are truly inspiring.

I am very thankful for the many friends NIE Jing, CHEN Calvin, XU Ceng, CHAU Wai Shing, CHEUNG Derek, HO Siu Wai, YANG Sheng Hao, FONG Salis, NGAI Michael, that I have had at the Chinese University of Hong Kong.

Very special thanks to my parents and family for their selfless love and support. I could not have accomplished this without them.

Abstract

在这篇论文中，我们研究TDMA系统的调度问题。我们的目的是在QoS的要求下，设计一个实际的算法去分配有限的无线网络资源。在这篇论文的头一部分，我们会探讨不同的无线网络模型和不同的功率分配方法。之后，我们会讨论两种不同的调度方法。我们也在每一种调度方法中，讨论一些算法。我们发现这些使用这两种调度方法的算法，都存在一些根本的问题，导致这些算法不能有效的分配资源。最后我们提出了一种调度方法和算法。通过仿真测试，我们证明我们提出的算法能有效解决那两种调度方法的问题。

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Summary

Wireless communication technology has been undergone dramatic changes in the past decades. In this chapter, we will provide an introduction of current wireless technologies, as well as the outline of the thesis.

1.1 Introduction to Wireless Technologies

In the past, sending a message with only few words to a place which is far away would easily take days for the delivery. Now, with the advent of wireless communication systems, unlimited information can be transmitted to anywhere almost instantly with only a click on a button. During the last decade, the technology of wireless communication has been under tremendous changes: from traditional analog voice messaging, to digital 2G systems, and to now robust high data-rate 3G and 3.5G systems as well as WiFi everywhere. These technology advancements include not only the increases in the speed and capacity of data transmissions, but also the enhancements in the reliability of the communication services. Due to the success of the wireless technology, people are expecting the technology to be further developed, with more capacity and better Quality-of-Service (QoS) comparable with the wired network. However, wireless spectrum is a scarce resource. Additional capacity cannot be provided through the installation of new physical devices like wired communication systems do. As wireless spectrum is a shared medium, the quality of service depends on the number of subscribers. The situation is different from wire-line communication systems in which each user has a dedicated channel and QoS is independent to the number of users. As a result, in order to satisfy the growing expectation of high data-rate and QoS, it is essential to allocate wireless spectrum properly. Therefore, the resource allocation problem becomes one of the biggest challenges for the development of wireless technology in the future.

1.2 Wireless Systems

Wireless systems compose of three main components: transmitters, receivers, and transmission medium (air). A signal message is converted to electromagnetic waves by the transmitter and broadcasted to the receiver through the air. When the receiver detects the signal waves, it would convert them back to the original signal message. The data conversion and broadcasting at both transmitters and receivers are performed by antennas. Generally, two types of antennas are commonly employed. They are omni-directional antennas and directional antennas.

Omni-directional antennas radiate signal power uniformly in all direction on a plane, while directional antennas radiate signal power in one directional sector. Depending on the distance the electromagnetic waves required to be propagated as well as the strength of the waves, the sizes of the antennas can be as big as the ones equipped in the base-stations, and can also be as small as the ones equipped in our handheld devices.

The strength of the broadcasting wave by the antenna directly affects the quality of transmission since the larger the power, the stronger the electromagnetic wave, the harder the waves get interfered. However, strong electromagnetic wave at the transmitter side does not guaranteed that the waves at the receiver side remain strong. The reason is that the signal waves would experience path loss, which is the reduction in the strength of signal power as the electromagnetic wave propagates. The loss is caused by the fact that the signal intensity is weaken when the electromagnetic wave gets through an object, which can be air, or a solid. The strength of signal power P at distance d away from the transmitter is estimated to be P/d^{α}, where α is called the path loss index and its value varies from different situations to different models. Typically, it is estimated that path loss index is ranging from 2-6 [16] [17]. In free space, where there is a direct line-of-sight as well as no obstacle between the sender and receiver, the path loss index is 2[15]; while in a city, the path loss index is estimated to be 4-6 [15], depending on the situation such as indoor or outdoor environments.

1.3 Wireless Networks

While wireless systems consist of transmitters and receivers, the configuration of these transceivers composes a wireless network. Generally, wireless network can be classified as two categories: Centralized networks and Ad-hoc networks.

Centralized Networks

In centralized wireless networks, devices do not communicate directly with each others but through a centralized hub. This

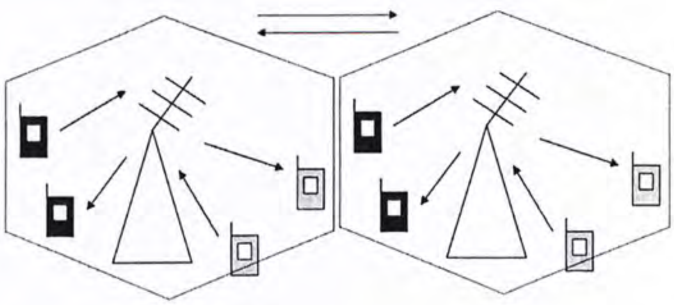

Figure 1.1: Centralized Network, where all transmission are processed through a fixed location wireless hub.

hub is usually fixed in location and serves as a router which redirects a message from a transmitter to a receiver. Thus, the hub controls the traffics of the network and the topology of the network is usually fixed. Cellular network[28] in telecommunication system is a kind of centralized networks, where a conversation between two end users is established through a base station. A base station in the cellular network is responsible to process all the transmissions within a certain range, forming a network autonomy called a cell. Two base stations communicates with each other if the destination of a transmission is not in the current cell. Figure 1.1 shows a typical centralized network.

Ad-hoc Networks

Ad-hoc network [27,29] has no centralized hub to direct messages to their destinations. In ad-hoc networks, wireless devices are randomly positioned, and therefore, ad hoc network has no fixed network topology. Two nodes can directly communicate with each other if they are within their own transmission ranges, unlike centralized network that every transmission has to be processed through the central hub. If a message is sent to the destination that is beyond the sender's range, the message would be

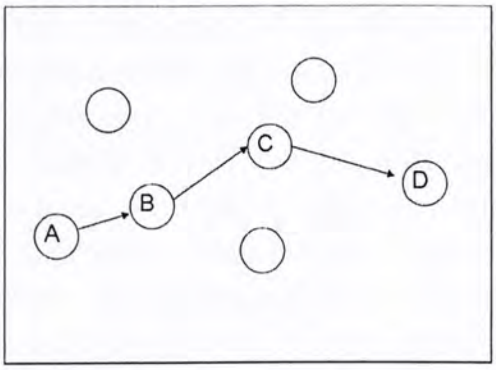

Figure 1.2: Ad Hoc Network, where a message is routed from transmitter A to receiver D through intermediate nodes B and C. Nodes B and C are mobile, and the path from node A and D is not always through B and C.

relayed to the intermediate nodes between the transmitter and the receiver. These intermediate nodes are mobile and therefore, the path between two devices is constantly changing depending on the mobility of the nodes in the network. Figure 1.2 is an example of ad hoc network.

1.4 Multiple Access

Since wireless spectrum is a shared medium, a scheme is required for users to share the spectrum. In wireless communication, multiple-access is a technique which allows multiple users to share a wireless medium simultaneously. Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), Code Division Multiple Access (CDMA) are the major techniques to allocate available bandwidth to multiple users.

TDMA

In TDMA systems, a radio spectrum is shared by multiple users, and each user occupies the spectrum a certain amount of time[30].

In other words, the time usage of the spectrum is divided into time slots. If a transmission cannot be finished in a time slot, it postpones its transmission and resumes in the later slots. The number of time slots required for a transmission depends on many factors, such as modulation scheme, amount of available bandwidth, and scheduling algorithms. Typically, for the fairness purposes, each user is guaranteed a chance to occupy the channel within a certain number of time slots, which is referred as a frame. For instance, a frame with n time slots would allow each user occupy the channel once. If a user cannot finish its transmission in a time slot, it would postponed its transmission to the next frame (which is n time slots later). In contract, the policy of time-slot allocation can also be based on priority. Instead of postponing to the next frame, consecutive time slots can also be assigned to a single user. Thus, bandwidth can be supplied on demand to different users, and this makes the resource allocation in the TDMA systems flexible. However, locating consecutive time slots on-demand requires tight synchronization since every user has to know when a slot starts and ends. The overhead of the transmission packet would become relatively large, and therefore, more bandwidth would be required.

FDMA

In FDMA systems, each user is assigned a unique frequency channel[30], unlike TDMA systems that multiple users are sharing the same channel. In other words, frequency spectrum is divided into frequency bands and assigned to users according to their requests. Transmissions in FDMA systems are continuous and the bandwidth of the channel is typically small (approximately 30kHz in cellular systems [15]) because each frequency channel only supports one user. Synchronization is not an issue in FDMA and the transmission overhead is relatively small compared with TDMA systems. However, as frequency is divided into channels, users in FDMA systems may suffer adjacent channel interferences which require tight radio frequency filtering. This is the major disadvantage of using FDMA systems.

CDMA

In CDMA systems, users share the same frequency band. A message would be encoded by correlating the message by a code sequence made up of chips, which is unique to each user. The code sequence would have a chip rate greater than the data rate of the message and the message is being "spread" over a larger bandwidth. The code sequences assigned to the users are constructed and they are approximately orthogonal to each others. The target receiver has to know its transmitter's code sequence in order to perform the de-correlation to retrieve the message (the undesired message would appear as noise under the de-correlation process). In other words, users in the CDMA systems occupy the shared frequency channel independently with their own code sequences.

Since a signal message is spreading over a large bandwidth, CDMA systems enjoy low multi-path fading. However, if the system allows too many users sharing the same radio frequency channel, it would result in near-far problem, which is a phenomenon that occurs when the signals nearby pose strong interferences (thus raising the noise floor) to the weak signals far-away, and therefore reduce the probability of the far-away weak signals to be received by the target receivers. This problem can be solved by proper power-control schemes.

1.5 Objectives and Outlines of the Thesis

In this thesis, scheduling problems of TDMA systems will be investigated. As discussed, proper time-slot allocation is critical in the TDMA systems. Our objective is to develop a scheduling algorithm which minimizes the number of time slots required to schedule a set of transmission requests. The organization of the thesis is as follow: In Chapter 2, we will discuss various scheduling criteria as well as different types of scheduling models in the TDMA systems. In Chapter 3, we will state the model and formulations used in the thesis. Subsequently, in Chapter 4, we will discuss different scheduling approaches in the nonlinear power control scheduling scheme, which is the scheme used in this thesis. In Chapter 5, we will introduce our algorithm, Step-Establish Algorithm (SEA), which is a nonlinear power control scheduling algorithm. In Chapter 6, we will investigate the performance of SEA by comparing it with two other nonlinear power control scheduling algorithms. Also, we will investigate the running time complexity SEA. Finally, in Chapter 7, we will conclude our work.

□ **End of chapter.**

Chapter 2

Background Studies

Summary

In this chapter, two classes of scheduling models will be discussed. They are interference-based model and graph-based model. Interference-based scheduling considers aggregated interference in a wireless network and a transmission is successful if its signal-to-noise ratio is greater than the transmission threshold, depending on the Quality-of-Service (QoS) requirement. In contrast, graph-based scheduling model considers that interference on a transmission is caused by neighboring transmitters within a certain range. Therefore, interference caused by the transmitters beyond this range is simply ignored.

In this thesis, interference-based model is employed. In this chapter, we will discuss different power control scheduling scheme in this model. It is shown that non-linear power assignment schemes are flexible and is an ideal scheme to obtain the optimal schedule.

2.1 Introduction of Scheduling Models of Wireless Networks (Graph-based and Interference-based)

The interference model in the TDMA scheduling systems can be generally divided into two categories: 1) Graph-based model [23], and 2) Interference-based model[31]. Graph-based scheduling model assumes that interference is caused by neighboring transmitters within certain distance and ignores the aggregated interference from transmitters beyond that distance. This scheduling model relies on a conflict graph, where nodes are connected if they pose strong interference on the transmission links that are typically within a range. In the conflict graph, the nodes in connection cannot be activated simultaneously in a time slot.

Typically, graph-based scheduling model considers two interference constraints:

Primary interference constraint, a node cannot transmit and receive simultaneously. In the other words, two links cannot be simultaneously active if they have a common node;

Secondary interference constraint, which refers to the interference on a receiver which is in the transmission range of other unintended transmitters. In the other words, two nodes can simultaneously transmit if and only if they are two hops away.

In Interference-based model, aggregated interference is considered and a channel can be reused as long as the signal-to-noise ratio SINR of a transmission is greater than its transmission threshold. This threshold is the minimum QoS level the user can accept. The SINR of a transmission greatly depends on power.

Interference-based model does not rely on conflict graph, since all nodes are considered to be conflicting with each other.

2.1.1 Discussion of the Two Scheduling Models

Graph-based scheduling model depends heavily on the conflict graphs, while different topologies with the same conflict graph would generally be considered identical. The intensity of interference is less concern. In addition, the primary and secondary interference constraints do not allow two overlapping links to be active at the same time, ignoring the fact that overlapping links are possibly be simultaneously active, resulting in additional time slots which are unnecessary. Graph-based scheduling model is usually used in ad hoc network and is relatively less frequently used in cellular network.[31]. Existing scheduling algorithms on conflict graphs in graph theory can be directly applied to this model.

In contrast, algorithms using interference-based scheduling model not only consider the topology of a network, but also power and SINR. Algorithms in this model may require more computational power than those in graph-based scheduling and this is the disadvantage of using this model. On the other hand, obtaining the optimal schedule using this model is NP-hard and heuristic algorithms are always used to approximate the optimal solution. In short, a trade-off between efficiency and computational complexity exists in this model.

2.1.2 Scheduling Model in the Thesis

In this thesis, interference-based scheduling model will be employed because we would like to develop an algorithm which gives a satisfactory result to any practical situations. Since graph-based scheduling model would be over-simplified network scenario, it is not a good choice in our thesis. In the subsequent section, we are going to discuss the criteria of using the interference-based scheduling model to develop a scheduling algorithm.

2.2 Power Assignment in Interference-based Scheduling Model

Graph- based scheduling protocols do not explicitly take power assignment into account but assume that each node is equipped with sufficient power for its transmission as well as that each node only interferes with its neighbors, which are the nodes that can have the direct transmissions to the sender node. On the other hand, in the Interference-based model, scheduling and power assignment are heavily associated. It is because the power level directly affects the SINR of a transmission. In the literatures, two approaches are generally taken for the power assignments. They are linear and nonlinear power assignments:

Linear power assignments[3], where each node transmits with power proportional to the minimum power used among all concurrent transmissions. Mathematically, $P_i = cP_{min}$, where c is a constant, P_i is power level by transmitter i, and P_{min} is the smallest power value among all transmitters. Linear power assignment, especially with c = 1, is frequently employed in multi-hop networks in the literature [18-22] since in multi-hop networks, transmitting devices are assumed to have similar properties and functionalities, such as transmission ranges and power consumptions. In addition, assuming the linearity of power levels would save the each user's computational power, which is a very limited resources in the multi-hop networks.

Non-linear power assignments, where the power level of each

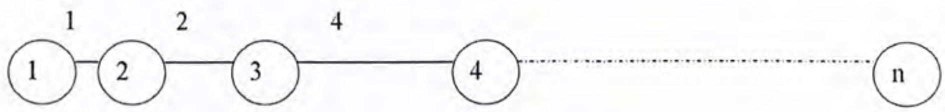

Figure 2.1: Network topology that [9] uses to show that scheduling protocols with linear power configurations would be inefficient compared to the non-linear power control scheduling protocols. In the network, each i and i + 1 are separated by 2^{i-1}.

sender is independent and can be adjusted to any value as long as the level is below the power limit and above the transmission threshold according to network situation. Networks with non-linear power assignment have better flexibility in the power level adjustments. However, transmitting devices in such networks may consume a significant portion of time in the computation of power configurations. Non-linear power assignment scheme is frequently employed in the literature on the topics of optimization of power consumption in the TDMA systems. [24]

As stated, the main benefit of using linear power assignment is its simplicity in computation compared to non-linear power configuration. However, there is always a tradeoff between efficiency and simplicity in the scheduling protocols. Moscibroda and Wattenhofer in [9] showed that algorithms using linear power assignment scheme may require a lot more time slots to schedule a set of transmission requests than the optimal value. They proved this statement in the network topology in Figure 2.1 and showed that in such topology, protocols with linear power assignment need a lot more time slots than protocols using nonlinear power assignment scheme.

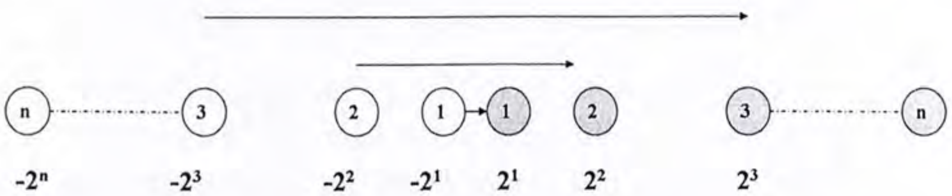

Figure 2.2: Exponential chain networks in which SRA is proved to schedule inefficiently

2.3 Motivation and Contribution

In general, in interferenced-based scheduling model, nonlinear power assignment scheme poses no limitation on the power consumption of wireless devices, making it the ideal model for efficient scheduling protocols. However, recently in [3], the authors argue that the current non-linear power control scheduling protocols are inefficient. In their work, they show that commonly-used step-removal algorithm, and its variations, requires n time slots to schedule n transmission requests on the network topology in Figure 2.2; while they proved that as the number of transmission requests n increases, the number of required time slots should increase with log(n) for the optimal schedule in this network topology. They also proposed an algorithm called low-disturbance scheduling (LDS) algorithm, which provides a near-optimal schedule in this network topology. However, the authors do not provided any in depth analysis of both algorithms in other network topologies. Motivated by [3], we would like to study the efficiencies of SRA and LDS, and also develop an efficient nonlinear power scheduling algorithm.

Our main contributions are as follow:

i) Detail analysis of SRA and LDS are performed. We find that both SRA and LDS have their own performance limitations on different network topologies.

ii) We will propose an algorithm which results in less time slots than SRA and LDS to schedule a set of transmission requests in all the network topologies studied in this thesis.

□ **End of chapter.**

Chapter 3

Model

Summary

In this chapter, we will state the model and definitions which are used throughout the thesis.

Consider a TDMA network with m devices, each of which is labeled by M_1, M_2,... M_m, interconnecting to each other through wireless links, which are labeled by l_1, l_2,.... A transmission link i is denoted as $l_i = (T(l_i), R(l_i))$, which represents a transmission from transmitter $T(l_i)$ to receiver $R(l_i)$, where $T(l_i)$, $R(l_i) \in \{M_1, M_2, \ldots, M_m\}$. The signal power on link i is denoted as $P(l_i)$. The channel gain from transmitter of link i to receiver of link j is denoted by $G(T(l_i), R(l_j))$. We consider the channel gain is due to the path loss, and is formulated as:

$$G(T(l_i), R(l_j)) = \frac{1}{d(T(l_i), R(l_j))^\alpha} \tag{3.1}$$

where $d(T(l_i), R(l_j))^\alpha$ is the distance from sender $T(l_i)$ of link i to receiver $T(l_j)$ of link j. α is the path loss index, which is a measure of the intensity of the path loss. The value of α is different from situations. In free space model, α is 2; while in city, α typically is assumed to be 4-6, depending on if direct line-of-sight(LOS), is possible.

$$\gamma(l_i) = \frac{P(l_i)G(T(l_i), R(l_i))}{\sum_{j:j\neq i} P(l_j)G(T(l_j), R(l_i)) + \eta} \tag{3.2}$$

where $\gamma(l_i)$ is the SINR of link i, and η is the thermal noise, which is assumed to be constant in this thesis. The SINR of each transmission link has to be greater than certain minimum level β in order to satisfy the Quality-of-Service (QoS). Therefore,

$$\gamma(l_i) \geq \beta(l_i) \tag{3.3}$$

In this thesis, we assume that each link has the same QoS requirement and minimum SINR level that has to reach. Thus,

$$\beta(l_i) = \beta(l_j) = \beta, \forall i, j \tag{3.4}$$

Definition 1: Denote $\wedge$ be the set of links with active transmission requests and L_t be the set of successfully scheduled requests in time t. Then a *schedule* is defined as $S(\wedge) = \{L_1, L_2, \ldots L_\tau\}$, where $\bigcup_{t=1}^{\tau} L_t = \wedge$ and $L_i \cap L_j = null \ \forall i \neq j$. In addition, we denote $|S(\wedge)|$=τ to be the number of time slots required to schedule all the transmission requests.

We assume that there is a scheduler in the TDMA system and the scheduler processes the transmission requests batch by batch. Incoming transmission requests are first waiting in a request pool, and then they are collected by the scheduler. If the scheduler collects the set of incoming transmission requests $\wedge_t$

from the pool at time t, then the next time the scheduler collects the incoming requests from the pool will be at time t + $|S(\wedge_t)|$. Further assume that in each batch, neither two transmitters transmit to the same destination, nor do two destinations receive messages from the same transmitter. This is equivalent to the primary interference constraint in the graph-based scheduling model.

From (3.2), if we denote

$$Z(l_i, l_j) = \frac{G(T(l_j), R(l_i))}{G(T(l_i), R(l_i))} \tag{3.5}$$

and

$$\eta / G(T(l_i), R(l_i)) = N(l_i) \tag{3.6}$$

We have

$$\gamma(l_i) = \frac{P(l_i)}{\sum_{j:j \neq i} P(l_j) Z(l_i, l_j) + N(l_i)} > \beta \tag{3.7}$$

Considering a set of n transmission requests which can be scheduled in a time slot, (3.7) would generate a matrix as (3.8)

$$\begin{bmatrix} P(1) \\ P(2) \\ \vdots \\ P(n) \end{bmatrix} \geq \beta \begin{bmatrix} Z(1,2) & Z(1,2) & \cdots & Z(1,n) \\ Z(2,1) & Z(2,2) & \cdots & Z(2,n) \\ \vdots & \vdots & \ddots & \vdots \\ Z(n,1) & Z(n,2) & \cdots & Z(n,n) \end{bmatrix} \begin{bmatrix} P(1) \\ P(2) \\ \vdots \\ P(n) \end{bmatrix} + \beta \begin{bmatrix} N(1) \\ N(2) \\ \vdots \\ N(n) \end{bmatrix} \tag{3.8}$$

Subsequently

$$(I - \beta Z)\overline{P} \geq \overline{N} \tag{3.9}$$

$$\overline{P} \geq (I - \beta Z)^{-1}\overline{N} \tag{3.10}$$

(3.10) provides us the necessary condition for the valid power assignment for the concurrent transmission requests. Since power has to be positive and also there may be a power limit for each device, we have a more contingent condition in (3.11)

$$\overline{P}_{limit} \geq \overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0 \tag{3.11}$$

where $\overline{P}_{limit}$ is the power limit vector. In this thesis we assume that there is no power limit and power can be adjusted to as large as infinite. Also, noise terms of all links are the same. Therefore,

$$N(l_i) = N(l_j) = N', \forall i, j \tag{3.12}$$

We determine the efficiency of an algorithm by its scheduling complexity, which is introduced in [3] and is re-defined as follows:

Definition 2: A scheduling complexity is a metric to measure the number of time slots required by an algorithm to schedule a set of transmission requests as the number of requests in the set increases.

An algorithm is said to be more efficient than another algorithm if the algorithm has lower scheduling complexity, and thus, requiring less time slots to schedule all the transmission requests. As for the short review, the following universal notation from complexity theory would be used to measure the efficient of an algorithm. Table 3.1 lists the notations used to measure the scheduling complexity of an algorithm. Table 3.2 lists the notations used in this thesis.

Table 3.1: **Notations for the scheduling complexity** ($f(n)$=$|S(\wedge)|$, the number of time slots required to schedule a set of transmission requests.)

$f(n) \in \Omega(\mathrm{n})$	$\exists(\mathrm{c} > 0,\ n_0{:}\forall(\mathrm{n}>n_0),|\mathrm{cn}|\leq|f(n)|)$
$f(n) \in O(\mathrm{n})$	$\exists(\mathrm{c} > 0,\ n_0{:}\forall(\mathrm{n}>n_0),|\mathrm{cn}|\geq|f(n)|)$
$f(n) \in O(log\ (\mathrm{n}))$	$\exists(\mathrm{c} > 0,\ n_0{:}\forall(\mathrm{n}>n_0),|\mathrm{c}log(\mathrm{n})|\geq|f(n)|)$
$f(n) \in O(log^2\ (\mathrm{n}))$	$\exists(\mathrm{c} > 0,\ n_0{:}\forall(\mathrm{n}>n_0),|\mathrm{c}log^2(\mathrm{n})|\geq|f(n)|)$

Table 3.2: Notations used throughout the thesis

m	Number of devices in a network
n	Number of transmission link requests in a network
l_i	Transmission link i
$T(l_i)$	Transmitter of transmission link i
$R(l_i)$	Receiver of transmission link i
$P(l_i)$	Transmission power of transmission link i
$G(i, j)$	Channel gain of transmitter i to receiver j
$d(T(l_i), R(l_j))$	Distance between transmitter of link i to receiver of link j. If j = i, $d(T(l_i), R(l_j))$ can be abbreviated to $d(l_i)$
$d(l_i)$	$=d(T(l_i), R(l_i))$, transmission distance between transmitter and receiver of link i
α	Path loss index
$\gamma(l_i)$	Signal-to-noise ratio (SINR) of transmission link i
β	Signal-to-noise ratio threshold
η	Thermal noise
$\wedge$	The set of all transmission requests
$S(\wedge)$	Schedule for the set of transmission requests $\wedge$
$Z(\cdot)$	Normalized channel gain

□ **End of chapter.**

Chapter 4

Nonlinear Power Assignment Scheduling Algorithm

Summary

In non-linear power scheduling scheme, two scheduling approaches are frequently employed. They are two-phase and step-removal approaches. In this chapter, we would discuss these two approaches. Also, we will investigate two algorithms, LDS and SRA, each of which belongs to one of the two scheduling approaches.

4.1 Nonlinear Power Control Scheduling Algorithms

Nonlinear power assignment scheme is the ideal approach for a protocol to achieve the optimal scheduling solution. However, finding the optimal schedule is a NP hard problem which requires significant amount of resources. Therefore, heuristic algorithms which are computationally-efficient and give suboptimal solution become one of the main focuses in the literatures [2], [8], [9]. The heuristic algorithms generally follow two main approaches: two-phase [10] and step-removal [1].

Two-Phase Approach

Two-phase approach is from Two-phase algorithm in [10]. Typically, in this approach, algorithms separate the scheduling process into the pre-scheduling phase and power control phase. In the pre-scheduling phase, the algorithm eliminates transmissions with senders that are within distance D to the current transmitter, with the purpose of eliminating "strong interferences". Thus, this phase is used to find the possible candidates for concurrent transmissions. In the power control phase, the algorithm determines if there is a possible power assignment to these concurrent transmission candidates. (i.e. to check if the power required for the concurrent transmissions exceeds the maximum power cap as well as below the minimum power limit.). The algorithm alternates these two phases iteratively until the requirements of them are satisfied. Two-phase scheduling algorithms are mostly applied in ad hoc network and have been implemented in many algorithms such as [3], [8], [9].

Step-Removal Algorithm

Step-removal approach is from Step-removal algorithm (SRA), which is a kind of greedy algorithm introduced by Zander [1]. The algorithm iteratively checks the scheduling condition and eliminates unwanted transmission links one by one. Therefore, it is a check-and-remove algorithm. Specifically, SRA first considers all the transmission requests as a group in the network and performs the following check-and-remove procedures:

i) Check if there is a power vector $\overline{P}$ satisfying the inequality (3.11). i.e. $\overline{P_{limit}} \geq \overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0$

ii) If no power vector is satisfied, remove the link which has the largest row/column sum of the normalized gain matrix Z, with the purpose of removing the link having the most difficulty to be scheduled.

The approach of SRA is widely used in the literatures and has been implemented to several variations. Zander, after SAR, proposed Limited Information Stepwise Removal Algorithm (LI-SRA), a linear power assignment version of SAR, which assumes senders with equal power and performs check-and-remove procedures as SRA. However, they are different in that LI-SRA removes the transmission request with the minimum SINR. Subsequently in [2], Lee implemented SRA and proposed Stepwise Maximum Interference Removal Algorithm (SMIRA), which removes the transmission link with the greatest aggregated interference $\{ \sum_{i \neq j} P_i Z_{ij}, \sum_{i \neq j} P_j Z_{ji} \}$ in (3.11), when power is assumed to be assigned optimally.

4.1.1 Efficiency of Two-Phase Algorithm

The efficiency of the two-phase algorithm depends heavily on the separation distance D in the pre-scheduling phase, while too large the distance D would over-eliminate possible concurrent transmission requests, resulting in inefficient schedule. In general, different network topologies have different optimal values of D, which makes the two-phase algorithm less practical. In addition, since different devices are experiencing different network situations, applying a fixed D as the separation distance between two transmitters throughout the whole network would inevitably over-eliminate possible concurrent requests, and greatly affect the performance of the algorithm.

4.1.2 Efficiency of Step Removal Algorithm

SRA does not perform any pre-scheduling link elimination that may over eliminates potential concurrent transmission requests. Thus, the scheduling complexity of SRA may typically be smaller than the scheduling complexity of the two-phase algorithms. However, [3] proves that it is not always the case. In [3], the author shows that using SRA and its variations could result in scheduling complexity which is significantly higher than the two-phase approach in some network topologies. They prove that for a network with topology as shown in Figure 4.1, using SRA would result in a scheduling complexity of $\Omega(n)$, while they provide a two-phase algorithm called Low-Disturbance Scheduling (LDS) algorithm which gives a schedule with complexity $O(log^2(n))$, and the optimal schedule is proved to have a complexity of $O(log(n))$. With this claim in mind, in the next section, we will investigate the LDS protocol and show that using LDS would also have a chance to result in significantly higher scheduling complexity than SRA. Therefore, both SRA and LDS are inefficient scheduling protocols.

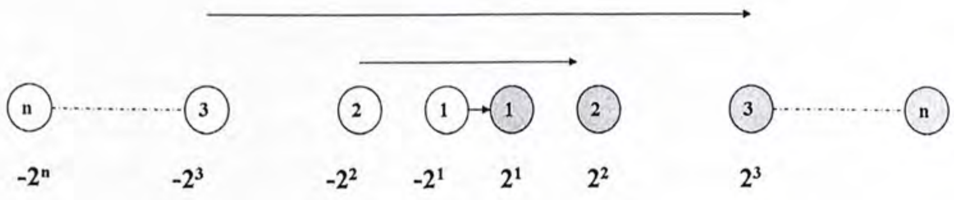

Figure 4.1: Transmitters and receivers are in white and black respectively. In this network, SRA schedules the link requests with complexity of $\Omega(n)$, while the optimal schedule with complexity $log(n)$ is possible.

4.2 Low-Disturbance Scheduling Protocol

Low-Disturbance Scheduling Protocol (LDS) is proposed by Moscibroda and Wattenhofer in [3] to illustrate the limitation of step-removal algorithm. LDS is a variation of two-phase algorithm It separates the scheduling into two phases, namely the pre-processing phase and the main scheduling phase. LDS is proved to give a schedule with complexity $O(log^2(n))$ in the topology in Figure 4.1, and requires at most $O(\chi_\rho \rho^2 log(n)(log(n) + \rho)$ time slots to schedule n transmission requests, where ρ is a constant that $\rho \geq 1$ and χ_ρ is called ρ-disturbance. The ρ-disturbance of a link l_i, denoted as $\chi_\rho(l_i)$, is the maximum number of senders or receivers within distance $d(l_i)/\rho$ of the transmitter or receiver of link l_i (i.e. for any link l_j, $|\{R(l_j)|d(R(l_j), R(l_i)) \leq d(l_i)/\rho\}|$ and $|\{T(l_j)|d(T(l_j), T(l_i)) \leq d(l_i)/\rho\}|$ [3], where $d(T(l_i), R(l_j))$ is the distance between transmitter $T(l_i)$ and receiver $R(l_j)$, and $d(l_i)$ is defined as $d(T(l_i), R(l_i)))$. Then, χ_ρ is defined as the maximum value of $\chi_\rho(l_i)$ for all l_i. The scheduling process of LDS is stated as following, and the detail algorithm and example are provided in Appendix B.

i) In the pre-processing phase, LDS considers transmission requests in the descending order of transmission lengths. The algorithm divides the n transmission requests $\wedge = \{l_1, l_2, \dots l_n\}$ into no more than $\lceil log(3n\beta) + \rho log(\alpha) \rceil$ groups according to their transmission distances, where β and α are SINR thershold and path loss index respectively. Each group has a group ID, γ, where $\gamma \in \{1, 2, \dots \lceil log(3n\beta) + \rho log(\alpha) \rceil$. Only transmission requests with the same group ID would be considered as candidates to be scheduled in the same time slot. In each group γ, LDS further divides the group into subgroups according to their transmissions distances. Each subgroup has a subgroup ID, τ. The pseudo code is listed below.(Notice that $\gamma(i)$ and $\tau(i)$ are the group ID and subgroup ID of link i respectively.)

Preprocessing Phase of LDS

Consider all links l_i in $\wedge$ in decreasing order of $d(l_i)$:
label links such that $d(l_i) \geq d(l_j)$ for $i<j$ and $i \neq j$
$\tau := 1$; $\gamma := 1$; *last* $:= d(l_1)$, the longest transmission length;
for each l_i in $\wedge$ **do**
    if last$/d(l_i) \geq 2$ **then**
        if $\gamma < \lceil log(3n\beta) + \rho log(\alpha) \rceil$ **then**
            $\gamma := \gamma + 1$;
        else
            $\gamma := 1$; $\tau := \tau + 1$;
        end
        last $:= d(l_i)$
    end
γ(i)$:= \gamma$; τ(i)$:= \tau$
end

ii) In the main scheduling phase, LDS examines the requests in each group γ and determines which requests are allowed to be transmitted by a subroutine called "allowed" subroutine. Since only requests with the same group ID are allowed to be scheduled simultaneously, the "allowed" subroutine determines which links are assigned the same time slot in a group γ by comparing the subgroup ID as well as the relative distances among different links. The pseudo code of the subroutine is stated in the following. (Notice that $\gamma(i)$ and

$\tau(i)$ are the group ID and subgroup ID of link i respectively.)

---

allowed subroutine

1: Define constant μ such that μ:=$4(120\beta(\alpha-1)/(\alpha-2))^{1/\alpha}$
2: **for** all the requests with the same group ID γ **do**
3: $\quad\delta_{ij} := \tau(i)$ - $\tau(j)$;
4: $\quad$**if** $\tau(i) = \tau(j)$ and $\mu d(l_i)\cdot > d(T(l_i), T(l_j))$
5: $\quad$**or** $\tau(i) > \tau(j)$ and $d(l_i) \cdot (3n\beta\rho^{\alpha})^{(\delta_{ij}+1)/\alpha} > d(T(l_i), R(l_j))$
6: $\quad$**or** $\tau(i) > \tau(j)$ and $d(l_j)/\rho > d(T(l_i), R(l_j))$
7: $\quad$**then return false**
8: **end for**
9: **return true**

Although LDS in [3] is proved to have a better efficiency than SRA in the network topology as shown in Figure 4.1, the comparison between these two algorithms is not fully studied in different topologies. In the following as well as the subsequent chapters, we will investigate the performance of both algorithms in different networks.

4.3 Fundamental Limitation of LDS

We have found that line 4 of the "allowed subroutine" causes a serious limitation on the performance of LDS. If a network with n requests is under the following conditions:

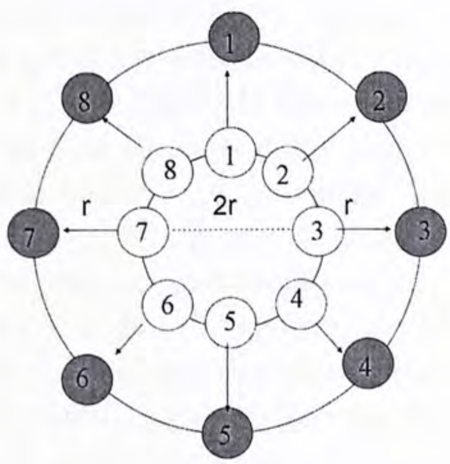

Figure 4.2: Ring topology with 8 transmissions separating two transmissions by 45 degree. Senders and receivers are in white and grey respectively.

Condition 1: $0.5 \leq d(l_i)/d(l_j) \leq 2, \forall i$

Condition 2: $d(T(l_i), T(l_j)) < \mu d(l_i), \forall i, j$, where $\mu = 4(120\beta(\alpha - 1)/(\alpha - 2))^{1/a}$

then LDS will always use n time slots to schedule these n links, while there may exist a schedule which needs less time slots. Consider a network shown in Figure 4.2 with 8 transmissions, where all transmission requests have the same transmission distance r. Transmitters and receivers are in white and grey, positioning on the rings with radius r and $2r$, respectively. The separation of two transmissions is 45 degrees.

According to LDS, since all eight transmission requests have the same transmission distances, they would be assigned the same group and sub-group values, with $\gamma = 1$ and $\tau = 1$ respectively. After the group ID and subgroup ID are assigned, the "allowed" subroutine is called to check if links can be scheduled simultaneously. With identical group and sub-group value for all the links, the conditions in line 4 of the "allowed" subroutine would be called. (i.e. to check whether $\mu \cdot d(l_i) > d(T(l_i), T(l_j))$, where $\mu = 4(120\beta(\alpha-1)/(\alpha-2))^{1/a}$) If this returns true, LDS would not schedule l_i and l_j in the same time slot, and vice versa. Assuming that $\beta = 2$ and $\alpha = 4$, μ will be approximately equal to 6.16 and $d(l_i) = r, \forall i$. Since all transmitters are placing on a ring, the longest distance between any two transmitters $d(T(l_i), T(l_j)), \forall i, j$ is $2r$. Therefore, $\mu d(l_i)$ is always larger than $d(T(l_i), T(l_j)), \forall i, j$, and the subroutine will always return false. Thus, no transmissions will be scheduled simultaneously in a time slot, and eight time slots are required to schedule all the transmission requests.

Next, we will show that only 1 time slot is needed for SRA to schedule all the transmission requests in the network. Consider the network topology in figure 4.2 with the same assumptions as the above. According to SRA, the algorithm would initially construct a 8x8 matrix Z, which is shown below in (4.1). After that, with the matrix Z, the algorithm checks if $\overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0$.

$$\beta Z = \begin{bmatrix} 0 & \frac{\beta}{2.17^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{2^{\alpha}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{2.17^{\alpha/2}} \\ \frac{\beta}{2.17^{\alpha/2}} & 0 & \frac{\beta}{2.17^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{2^{\alpha}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} \\ \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{2.17^{\alpha/2}} & 0 & \frac{\beta}{2.17^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{2^{\alpha}} & \frac{\beta}{7.83^{\alpha/2}} \\ \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{2.17^{\alpha/2}} & 0 & \frac{\beta}{2.17^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{2^{\alpha}} \\ \frac{\beta}{2^{\alpha}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{2.17^{\alpha/2}} & 0 & \frac{\beta}{2.17^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{7.83^{\alpha/2}} \\ \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{2^{\alpha}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{2.17^{\alpha/2}} & 0 & \frac{\beta}{2.17^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} \\ \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{2^{\alpha}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{2.17^{\alpha/2}} & 0 & \frac{\beta}{2.17^{\alpha/2}} \\ \frac{\beta}{2.17^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{2^{\alpha}} & \frac{\beta}{7.83^{\alpha/2}} & \frac{\beta}{5^{\alpha/2}} & \frac{\beta}{2.17^{\alpha/2}} & 0 \end{bmatrix} \tag{4.1}$$

With $\beta = 2$ and $\alpha = 4$, the matrix assumes the numerical values of Z in (4.2). According to the Perron-Frobenius Theorem, for $\overline{P} \geq (I - \beta Z)^{-1}\overline{N}$, if the maximum row/columnn sum of $\beta Z < 1$, there exist a positive power vector $\overline{P}$ such that $\overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0$. Therefore, if there is no power limitation, all 8 transmission requests can be scheduled concurrently in a time slot, resulting in that only 1 channel is needed.

In this example, we actually want to show that in the networks scenario as Figure 4.3 which has the properties of Condition 1 and 2 , LDS needs to use n time slots to schedule the n transmission requests, even though the optimal number of required time slots is far less than n. In the next chapter, we will provide simulations to further support this claim.

$$\beta Z = \begin{bmatrix} 0 & 0.4241 & 0.08 & 0.0326 & 0.125 & 0.0326 & 0.08 & 0.4241 \\ 0.4241 & 0 & 0.4241 & 0.08 & 0.0326 & 0.125 & 0.0326 & 0.08 \\ 0.08 & 0.4241 & 0 & 0.4241 & 0.08 & 0.0326 & 0.125 & 0.0326 \\ 0.0326 & 0.08 & 0.4241 & 0 & 0.4241 & 0.08 & 0.0326 & 0.125 \\ 0.125 & 0.0326 & 0.08 & 0.4241 & 0 & 0.4241 & 0.08 & 0.0326 \\ 0.0326 & 0.125 & 0.0326 & 0.08 & 0.4241 & 0 & 0.4241 & 0.08 \\ 0.08 & 0.0326 & 0.125 & 0.0326 & 0.08 & 0.4241 & 0 & 0.4241 \\ 0.4241 & 0.08 & 0.0326 & 0.125 & 0.0326 & 0.08 & 0.4241 & 0 \end{bmatrix} \tag{4.2}$$

4.4 Chapter Conclusion

In this chapter, we have investigated two different wireless scheduling approaches: step-removal and two-phase. We have also discussed that protocols using these approaches will have limitations on different network topologies: Step-Removal Algorithm has high scheduling complexity in exponential chain network; while although Low-Disturbance Scheduling Protocol results in

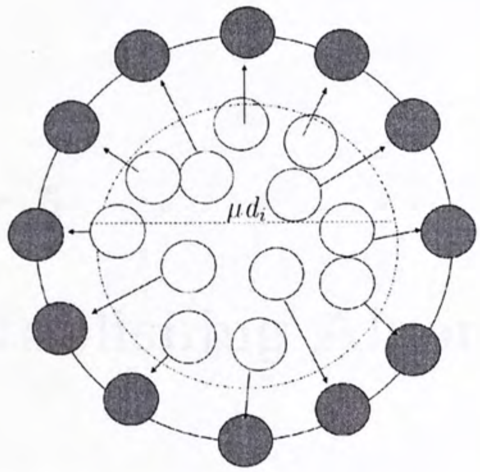

Figure 4.3: Network topology which LDS would schedule inefficiently.

efficient schedule in exponential chain network, if a network has transmissions which have similar transmission length as well as the senders of these transmissions are in close proximity, LDS will result in high scheduling complexity. Motivated by these, in the next chapter, we would like to propose the step-establishing scheduling approach which gives robust performance compared to the two-phase and step-removal approaches.

□ **End of chapter.**

Chapter 5

Step-Establishing Algorithm

Summary

Step-Establishing Algorithm (SEA) is a check-and-establish algorithm. It is similar to step-removal algorithm but the crucial establishing condition is different from the removal condition in SRA. In this chapter, we will introduce this algorithm in detail and in the next chapter, we will provide simulations to illustrate the efficiency of SEA.

5.1 Step-Establishing Algorithm

In Chapter 4, it is shown that step-removal and two-phase scheduling approaches have limitations on certain network topologies. In this chapter, we will propose a nonlinear power control scheduling protocol called Step-Establishing Algorithm (SEA). The idea of SEA is derived from SRA and has the following properties:

i) Similar to SRA, SEA has two stages, the checking stage and the scheduling stage.

ii) Instead of using normalized gain matrix Z, in the checking stage, SEA employs a matrix Φ, which is proposed in this thesis and is defined as $\Phi = \beta^2 Z \circ Z^T$, where $\circ$ is the hadamard product operation, to determine which links are chosen to be scheduled.

iii) Instead of removing the link with maximum row/column, SEA adds a link which results in the minimum sum of all the entries of the corresponding Φ.

The checking stage is critical to both SRA and SEA. In the following example, we will illustrate the reasons of proposing the matrix Φ as well as a new selection method instead of using the row/column sum method on matrix Z as in SRA.

Example I-a: Illustration of why the row/column on Z is not used in SEA

$$Z = \begin{bmatrix} 0 & 0.2 & 0.1 & 2 \\ 0.2 & 0 & 1 & 1.5 \\ 0.8 & 1.5 & 0 & 0.6 \\ 1 & 1.5 & 0.2 & 0 \end{bmatrix} \tag{5.1}$$

Consider a network with 4 transmission requests which yield the matrix Z in (5.1), where each link i corresponds to row/column i. First of all, we will show that the minimum number of time slots to schedule these four links is two. Recalled that if all links can be scheduled in the same time slot, $\overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0$ should be satisfied. With matrix Z in (5.1), $(I - \beta Z)^{-1}$ will have values in (5.2), with $\beta = 2$ as before.

$$(1 - \beta Z)^{-1} = \begin{bmatrix} -4.412 & -0.294 & 10.294 & -11.765 \\ 2.157 & -0.245 & -4.963 & 4.988 \\ 1.471 & -0.735 & -2.390 & 2.463 \\ -0.980 & -0.343 & 1.801 & -2.267 \end{bmatrix} \tag{5.2}$$

For illustration purpose, in this example, we assume that the normalized noise term of every link is positive constant N'.

(Thus, $\overline{N} = [N' \ N' \ N' \ N']^T$). Therefore, $(I - \beta Z)^{-1}\overline{N} = [-6.177N' \ 1.937N' \ 0.809N' \ -1.789N']^T$. Since power has to be positive, $\overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0$ does not hold. As the result, more than one time slots are needed to schedule these four links.

The possible optimal grouping which results in a schedule with two time slots is shown in Figure 5.1, with link 1 and 2 as a group and link 3 and 4 as another group. We will prove this shortly.

$$
\begin{array}{cc}
 & \begin{array}{cccc} \text{link 1} & \text{link 2} & \text{link 3} & \text{link 4} \end{array} \\
\begin{array}{c} \text{link 1} \\ \text{link 2} \\ \text{link 3} \\ \text{link 4} \end{array} &
\left[\begin{array}{cc|cc} 0 & 0.2 & 0.1 & 2 \\ 0.2 & 0 & 1 & 1.5 \\ \hline 0.8 & 1.5 & 0 & 0.6 \\ 1 & 1.5 & 0.2 & 0 \end{array} \right]
\end{array}
$$

Figure 5.1: The optimal grouping of matrix in (5.1)

First of all, it can be seen that $Z_{\{l_1,l_2\}}$, the corresponding Z for group with link 1 and 2, equals to

$$
Z_{\{l_1,l_2\}} = \begin{bmatrix} 0 & 0.2 \\ 0.2 & 0 \end{bmatrix} \tag{5.3}
$$

With the normalized noise for link 1 and 2, $\overline{N}_{\{l_1,l_2\}} = [N' \ \ N']^T$ and the numerical assumptions as before,

$$
(I - \beta Z_{\{l_1,l_2\}})^{-1} = \begin{bmatrix} 1.563 & -0.938 \\ -0.938 & 1.563 \end{bmatrix} \tag{5.4}
$$

resulting in $(I - \beta Z_{\{l_1,l_2\}})^{-1}\overline{N}_{\{l_1,l_2\}} = [0.625N' \ \ 0.625N']^T$ and therefore, $\overline{P}_{\{l_1,l_2\}} \geq (I - \beta Z_{\{l_1,l_2\}})^{-1}\overline{N}_{\{l_1,l_2\}} > 0$ is possible.

On the other hand, $Z_{\{l_3,l_4\}}$, the corresponding Z for group with link 3 and 4, equals to

$$Z_{\{l_3,l_4\}} = \begin{bmatrix} 0 & 0.6 \\ 0.2 & 0 \end{bmatrix} \tag{5.5}$$

With $\overline{N}_{\{l_3,l_4\}} = [N' \ \ N']^T$, resulting in $(I - \beta Z_{\{l_3,l_4\}})^{-1}\overline{N}_{\{l_3,l_4\}} = [1.072N' \ \ 0.357N']^T$. Therefore, both groups are valid grouping and minimum two time slots are required to schedule all the links.

Next, we will show that SRA requires three time slots to schedule all the links.

i) Since link 1, 2, 3, 4 cannot be simultaneously scheduled in the same time slot, SRA will remove the link with the maximum row/column sum, which is link 2 with greatest sum 3.2.

ii) The matrix $Z_{\{l_1,l_3,l_4\}}$ for the remaining link 1, 3, 4 will be

$$Z_{\{l_1,l_3,l_4\}} = \begin{bmatrix} 0 & 0.1 & 2 \\ 0.8 & 0 & 0.6 \\ 1 & 0.2 & 0 \end{bmatrix} \tag{5.6}$$

With normalized noise vector for link 1, 3, 4, $\overline{N}_{\{l_1,l_3,l_4\}} = [N' \ \ N' \ \ N']^T$, $(I-\beta Z_{\{l_1,l_3,l_4\}})^{-1}\overline{N}_{\{l_1,l_3,l_4\}}$ will be $[-8.5N' \ \ -5N' \ \ -4.5N']^T$. Therefore, no power vector $\overline{P}_{\{l_1,l_3,l_4\}} \geq (I - \beta Z_{\{l_1,l_3,l_4\}})^{-1}\overline{N}_{\{l_1,l_3,l_4\}} > 0$, and link 1, 3, 4 cannot be scheduled in the same time slot. Link 4 will be removed since it has the greatest column sum of 2.6.

iii) The corresponding Z for link 1 and 3 is

$$Z_{\{l_1,l_3\}} = \begin{bmatrix} 0 & 0.1 \\ 0.8 & 0 \end{bmatrix} \tag{5.7}$$

and with $\overline{N}_{\{l_1,l_3\}} = [N' \;\; N']^T$, $(I - \beta Z_{\{l_1,l_3\}})^{-1}\overline{N}_{\{l_1,l_3\}}$ will be $[0.136N' \;\; 1.081N']^T$, and therefore, positive power vector for link 1, 3 is possible. Thus, both links can be scheduled in the same time slot.

iv) It can be easily seen that link 2 and 4 cannot be scheduled simultaneously. Therefore, they are scheduled in different time slots. The resulting grouping of links is shown in Figure 5.2.

$$
\begin{array}{c}
\\
\text{link 1} \\
\text{link 3} \\
\text{link 2} \\
\text{link 4}
\end{array}
\begin{array}{c}
\begin{array}{cccc}
\text{link 1} & \text{link 3} & \text{link 2} & \text{link 4}
\end{array} \\
\left[\begin{array}{cccc}
0 & 0.1 & 0.2 & 2 \\
0.8 & 0 & 1 & 0.6 \\
0.2 & 1 & \boxed{0} & 1.5 \\
1 & 0.2 & 1.5 & \boxed{0}
\end{array}\right]
\end{array}
$$

Figure 5.2: The grouping of the matrix in (5.1) by SEA using matrix Z

Finally, we will show that applying the row/column sum logic to SEA will result in high scheduling complexity which is similar to the case of SRA. Since SEA adds one link at a time instead of removing links one by one, if applying the row/column sum method, SEA would add a link with the minimum resulting row/column sum. Below is how SEA operates with this method.

i) Start with link 1, since link 1 has the minimum row/column sum in (5.1)

ii) Then, link 3 will be added to the group of link 1 because the resulting matrix $Z_{\{l_1,l_3\}}$ equals to,

$$Z_{\{l_1,l_3\}} = \begin{bmatrix} 0 & 0.1 \\ 0.8 & 0 \end{bmatrix} \tag{5.8}$$

which has the minimum row sum of 0.1.

iii) No further link would be added to the group of link 1 and link 3 because it can be seen from the above that link 1, 3, 2, cannot be scheduled in the same time slot. Also, link 1, 3, 4 cannot be scheduled concurrently. Therefore, link 2 and 4 would be scheduled in the latter time slots.

iv) Since link 2 and 4 cannot be scheduled simultaneously, they would be scheduled in the different time slots, resulting in the grouping in Figure 5.2. Therefore, three time slots would be used if the logic of row/column sum is applied to SEA.

Later in this chapter, we will show that using our proposed method as a selection condition will only need two time slots to schedule all the links in this example. Before this, we will explain the algorithm in the next subsection.

5.1.1 Conditions of Link-Establishment in SEA

Basically, Φ is a matrix which provides the information of which two links can be scheduled. If the (i, j) entry of Φ is greater than or equal to 1, then the link i and j cannot be scheduled simultaneously. The following is derivation of Φ.

Consider a network with two transmissions requests l_1 and l_2 shown in Figure 5.3 below.

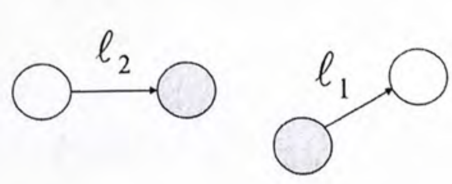

Figure 5.3: Random network with two transmissions.

If both transmission requests are to be scheduled simultaneously, it is required that the signal-to-noise ratio SINR of each link request above the SINR threshold. For transmission 1, that is:

$$\frac{P(l_1)G(l_1,l_1)}{P(l_2)G(l_2,l_1)+\eta} = \frac{P(l_1)}{z_{12}P(l_2)+N'} \geq \beta \tag{5.9}$$

$$P(l_1) \geq \beta z_{12}P(l_2) + \beta N' \tag{5.10}$$

where P, η, β are power, thermal noise, and the SINR threshold respectively. $G(l_i, l_j)$ is the channel gain between link i and j. $N' = \eta/G(l_1, l_1)$ and $z_{ij} = G(l_j, l_i)/\ G(l_i, l_i)$. Similarly, for transmission 2, it is

$$\frac{P(l_2)G(l_2,l_2)}{P(l_1)G(l_1,l_2)+\eta} = \frac{P(l_2)}{z_{21}P(l_1)+N''} \geq \beta \tag{5.11}$$

$$P(l_2) \geq \beta z_{21}P(l_1) + \beta N'' \tag{5.12}$$

where $N'' = \eta/G(l_2, l_2)$. Solving the power requirement for the inequalities of (5.11) and (5.12) if both links are to be scheduled in the same time slot, we will have the following power requirements.

$$P(l_1) \geq \frac{\beta^2 z_{12}N'' + \beta N'}{1-\beta^2 z_{12}z_{21}} \tag{5.13}$$

$$P(l_2) \geq \frac{\beta^2 z_{21}N' + \beta N''}{1-\beta^2 z_{21}z_{12}} \tag{5.14}$$

Since power has to be positive, the denominators of (5.13) and

(5.14), which both equal to $1-\beta^2 z_{12}z_{21}$, have to be greater than 0. Denote $\varphi_{12}=\beta^2 z_{12}z_{21}$ and $\varphi_{21}=\beta^2 z_{21}z_{12}$respectively, It can be seen that $\varphi_{12}=\varphi_{21}<1$.

Remark 1: Two transmission requests l_i and l_j are allowed to be scheduled concurrently in a time slot if and only if $\varphi_{ij}<1$, where $\varphi_{ij}=\varphi_{ji}=\beta^2 z_{ij}z_{ji}$, β is the SINR threshold, and z_{ij} is the normalized channel gain $G(l_j,l_i)/\ G(l_i,l_i)$.

The physical meaning of φ_{ij} is that the closer the value of φ_{ij} to 1, the harder the two transmissions l_i and l_j be scheduled in the same time slot since power required would be very large approaching to infinite. Therefore, as in contrast to LDS that using ρ- disturbance, we use φ as a measure of the fundamental difficulties of scheduling two link requests.

Remark 2:Denote Φ to be a matrix with entry φ_{ij}, and each entry φ_{ij} is defined as $\beta^2 z_{ij}z_{ji}$, then $\Phi=\beta^2 Z\circ Z^T$, where z_{ij} is the (i,j) entry of Z and $\circ$ is the hadamard product operation. Also, $\varphi_{ij}=\varphi_{ji}=\beta^2 z_{ij}z_{ji}=\beta^2 z_{ji}\cdot z_{ij}$.

Remark 3: For $0\leq\varphi_{ij}<1$, the larger the value of φ_{ij}, the more difficult link i and j to be scheduled simultaneously in the same time slot, and vice versa.

Definition 3: Denote $\wedge$ as the set of all the links with active requests which results in matrix Φ. If A is the subset of $\wedge$, Φ_A is the submatrix of Φ consists of links listed in the subset A.

Remark 4: Let L be the set of successfully scheduled requests and Φ_L be the corresponding matrix Φ for links in L. If there exist two links, l_i and l_j, waiting to be scheduled, each of which can be simultaneously scheduled with all the links in the set L, then l_i is chosen to be added to L if and only if $\Phi_{L\bigcup l_i}$ has smaller total entry sum than $\Phi_{L\bigcup l_i}$.

If $\Phi_{L\bigcup l_i}$ has the smaller total entry sum than l_i $\Phi_{L\bigcup l_i}$, l_i should have the smaller impact of the overall scheduled links. Therefore, l_i is chosen over l_j. Thus, Remark 4 is the critical selection condition for SEA.

Table 5.1 provides the notations used in the algorithm, which is stated as Algorithm 1 in the next page.

Table 5.1: Notations used in the Step-Establishing Algorithm

l_i	Transmission link i
$\wedge$	The set of all links with active transmission requests.
L_t	The set of successfully scheduled requests in time slot t
$S(\wedge)$	Schedule of all transmission links. $S(\wedge) = \{L_1, L_2, \ldots L_\tau\}$. The number of required time slots to schedule all the links is τ $\lvert S(\wedge)\rvert = \tau$.
$\varphi_{ij} = \beta^2 z_{ij} z_{ji}$	The condition that for two links to be scheduled in a time slot If $\varphi_{ij} < 1$, l_i and l_j can be scheduled in the same time slot.
Φ	Matrix with entry Φ(i, j) $= \varphi_{ij}$

Example I-b: Illustration of SEA

Recalled Example I-a that 4 transmission requests are waiting to be scheduled, with matrix Z shown in (5.1). The matrix Φ of all the four links is shown in (5.15). Denote L_t be the set of successfully scheduled links in time slot t. SEA will schedule these links in the following.

$$\Phi = \begin{bmatrix} 0 & 0.16 & 0.32 & 4 \\ 0.16 & 0 & 4 & 9 \\ 0.32 & 4 & 0 & 0.16 \\ 4 & 9 & 0.16 & 0 \end{bmatrix} \tag{5.15}$$

i) At time slot $t = 1$, $L_1 =$ null. SEA will first start with the link with the minimum row/column sum of Φ, with the purpose of picking the link with the least restriction in the network. Link *1* and link *3* both have the smallest row/column sum of 4.48, and therefore, either one can be chosen. We randomly pick link *1*. Thus, $L_1 = \{l_1\}$.

ii) The algorithm, then, picks the link i which has the minimum sum of all the entries in the resulting $\Phi_{l_1 \bigcup \{l_i\}}$. In this case, $i = 2$ since $\Phi_{\{l_1, l_2\}}$ equals to the following:

$$\Phi_{\{l_1,l_2\}} = \begin{bmatrix} 0 & 0.16 \\ 0.16 & 0 \end{bmatrix} \tag{5.16}$$

and the matrix has the minimum corresponding matrix total entry sum of 0.32.

iii) The algorithm checks if there exists a power vector $\overline{P}_{\{l_1,l_2\}}$ such that $\overline{P}_{\{l_1,l_2\}} \geq (I - \beta Z_{\{l_1,l_2\}})^{-1}\overline{N}_{\{l_1,l_2\}} > 0$, where $Z_{\{l_1,l_2\}}$ is the corresponding Z for link *1* and *2*, and $\overline{N}_{\{l_1,l_2\}} = [N' \ N']^T$. (We have shown previously that there exists such vector). Thus, $L_1 = \{l_1, l_2\}$

iv) Although $\Phi_{\{l_1,l_2,l_3\}}$ has smaller total entry sum than $\Phi_{\{l_1,l_2,l_4\}}$, link *3* will not be added because $\Phi_{\{l_1,l_2,l_3\}}$ contains entries which are greater than 1. Similarly, link *4* will not be added.

v) Now, t = 2. L_2 will start with a null set, and finally, link *3* and *4* will be added. Thus, $L_2 = \{l_3, l_4\}$. The resulting grouping is shown in the Figure 5.4.

$$\Phi = \begin{array}{c} \\ \text{link 1} \\ \text{link 2} \\ \text{link 3} \\ \text{link 4} \end{array} \begin{array}{c} \begin{array}{cccc} \text{link 1} & \text{link 2} & \text{link 3} & \text{link 4} \end{array} \\ \begin{bmatrix} 0 & 0.16 & 0.32 & 4 \\ 0.16 & 0 & 4 & 9 \\ 0.32 & 4 & 0 & 0.16 \\ 4 & 9 & 0.16 & 0 \end{bmatrix} \end{array} \qquad Z = \begin{array}{c} \\ \text{link 1} \\ \text{link 2} \\ \text{link 3} \\ \text{link 4} \end{array} \begin{array}{c} \begin{array}{cccc} \text{link 1} & \text{link 2} & \text{link 3} & \text{link 4} \end{array} \\ \begin{bmatrix} 0 & 0.2 & 0.1 & 1 \\ 0.2 & 0 & 1 & 1.5 \\ 0.8 & 1 & 0 & 0.2 \\ 1 & 1.5 & 0.2 & 0 \end{bmatrix} \end{array}$$

Figure 5.4: The grouping by SEA for the matrix in (5.1)

Algorithm 1: Step-Establishing Algorithm

for n requests $\wedge = \{l_1, l_2, \ldots, l_n\}$
time slot t := 1 ;
while there are links to be scheduled do
    L_t= null
    //L_t is the set of successful scheduled links
    //pick the first link for the group L_t
    //$\Phi_\wedge$ = the Φ matrix for links in $\wedge$
    find l_i with minimal row/column sum of $\Phi_\wedge$,
    add l_i to L_t
    //remove l_i from unscheduled set $\wedge$
    $\wedge \longleftarrow \wedge \backslash l_i$
    while there are links to be scheduled
        find all the links where each of these links l_j satisfies
        that matrix $\Phi_{L_t \cup l_j}$ contains no entry ≥ 1
        Group these links to C_t // possible candidates to be scheduled
        while C_t is not empty
            find $l_{min} \in C_t$such that the sum of all entries in
            $\Phi_{L_t \cup l_{min}}$ is minimal
            check if there exists a power vector for
            inequality $P \geq (I - \beta Z_{L_t \cup l_{min}})^{-1}\overline{N}$
            if there exists a power vector
                add l_{min} to L_t, and $\wedge \longleftarrow \wedge \backslash l_{min}$
            if no possible power vector exists
                $C_t \longleftarrow C_t \backslash l_{min}$
        end while
        t = t + 1
    end while
end while

□ **End of chapter.**

Chapter 6

Performances of LDS, SRA, and SEA

Summary

In this chapter, the performances of three algorithms, LDS, SRA, and SEA would be compared under simulations.

6.1 Simulation

i) LDS, SRA, SEA are compared under simulations.

ii) Simulations are performed on MATLAB.

iii) Network topologies are first generated and the coordinates (x, y) of all communication devices are stored in matrix. The coordinate matrix also stores the and thermo noise information of transmission pairs.

iv) Coordinates of all devices are unique, meaning that there is no overlapping nodes.

v) The input of each algorithm is the coordinate matrix.

vi) Constant parameters are: SINR threshold $\beta = 2$, path loss index $\alpha = 4$, $\eta = 1$.

6.2 Exponential Chain Topology

Purposes

[3] has proved that that SRA needs n time slots to schedules n transmission requests in the network topology shown in Figure 6.1. It also proposes LDS, which would give lower scheduling complexity than SRA. In this section, we mainly would like to compare the performance of SEA and LDS, and to see if SEA would result in better schedule.

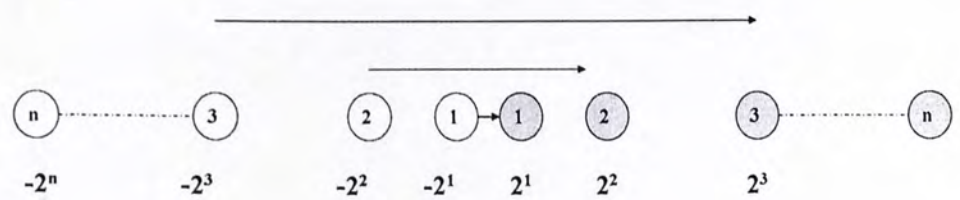

Figure 6.1: Exponential chain networks in which SRA is proved to schedule inefficiently

Scopes

We will simulate n = 10 to 500 numbers of transmission requests.

Simulation Results

The simulation result is shown in Figure 6.2. It can be see that LDS requires more time slots than SEA to schedule these n requests. In addition, as the number of requests increases, the number of required time slots to schedule these requests increases more rapidly for LDS than for SEA. As the result, it can be conculded that SEA has lower scheduling complexity than LDS, and therefore, is the most efficient algorithm among LDS and SRA in this topology.

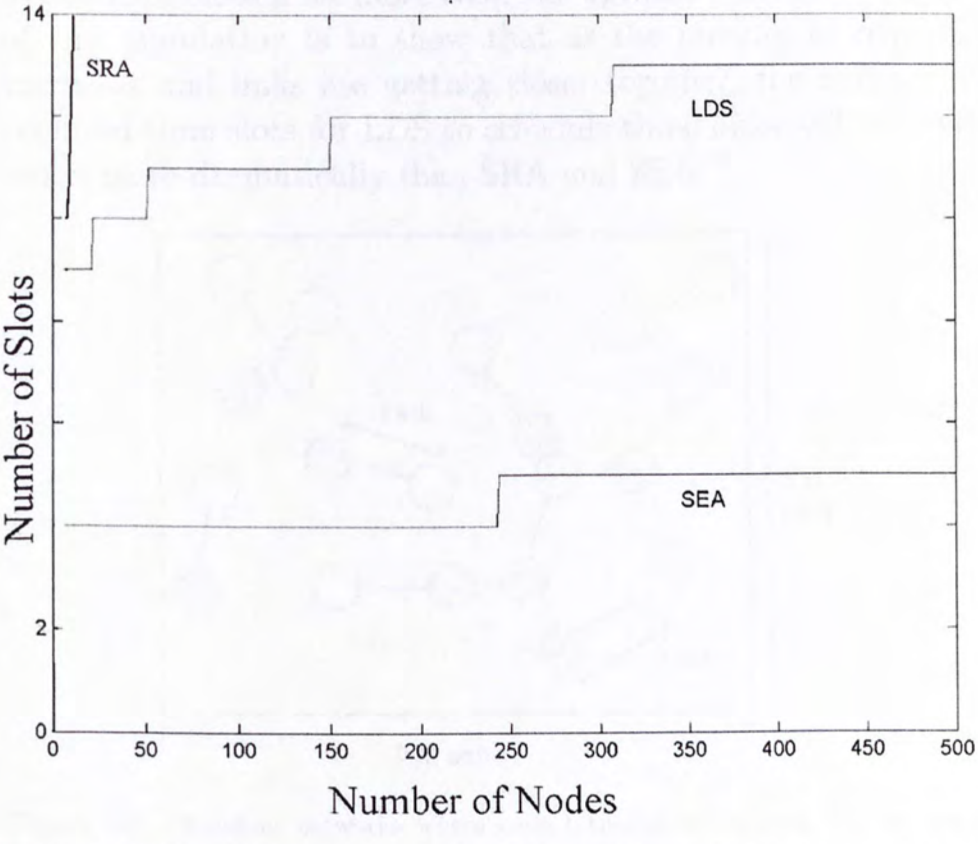

Figure 6.2: Simulation result of number 10 to 500 transmissions on exponential chain.

6.3 Fixed-Transmission-Length Random Network

Purposes

In this simulation, all transmission requests have the same transmission length and are randomly distributed on a plane, which is shown in Figure 6.3. In Chapter 4, we have shown that if a network with properties that I) $0.5 \leq d(l_i)/d(l_j) \leq 2$, $\forall i$, and II) $d(T(l_i), T(l_j)) < \mu d(l_i)$, $\forall i, j$, where $\mu = 4(120\beta(\alpha-1)/(\alpha-2))^{1/a}$, then LDS may result in schedules which require the number of time slots a lot more than the optimal one. The purpose of this simulation is to show that as the number of requests increases and links are getting closer together, the number of required time slots for LDS to schedule these links will increase much more dramatically than SRA and SEA. .

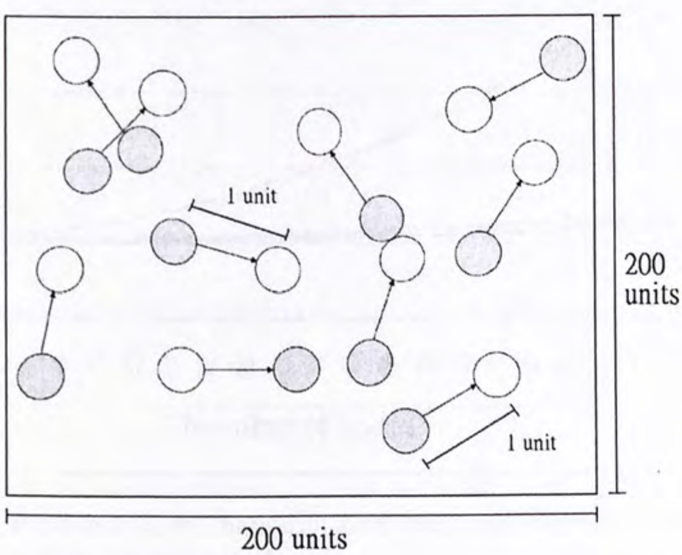

Figure 6.3: Random networks where each transmission request has the same transmission length.

Scopes

The dimension of the plane is 200x200 $unit^2$. The number of transmission requests in this simulation are from 1 to 90. Each transmission link has transmission distance of 1 unit, i.e. $d(l_i)$ = 1 unit, $\forall i$. The coordinates of senders nodes are first generated randomly by the MATLAB random function, and then the receivers are randomly placed 1 unit away of their corresponding senders. For each specific number of requests n, 500 randomly-generated network scenarios are simulated, and the results are to be averaged. The averaged results are plotted in Figure 6.4.

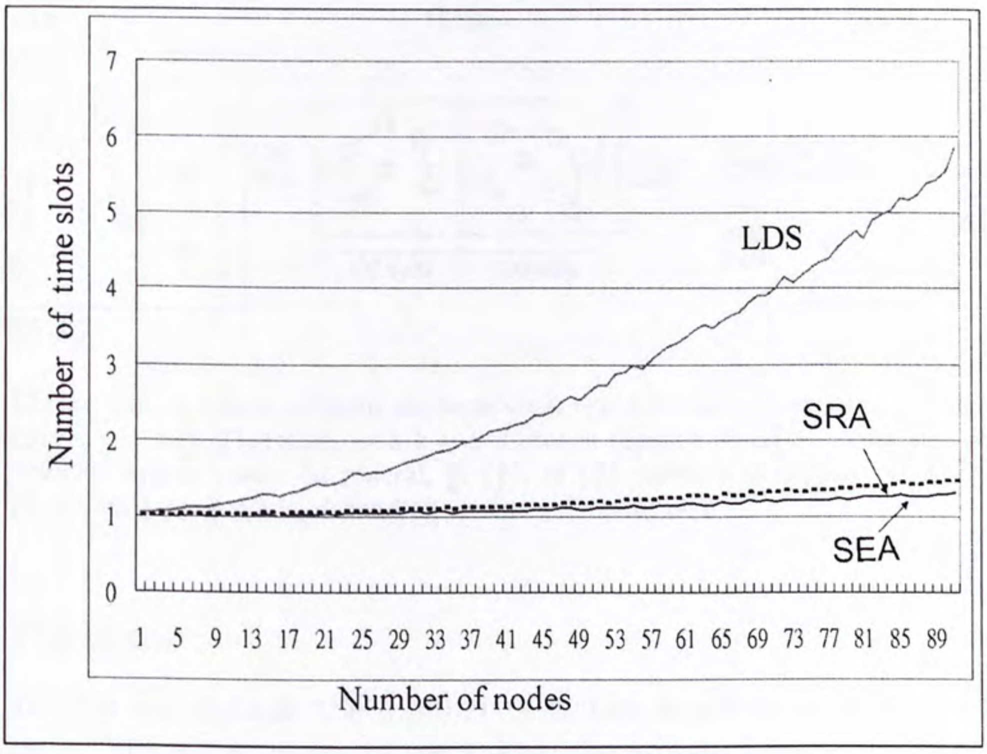

Figure 6.4: Performance evaluation of LDS, SRA, and SEA in random network for requests with fix transmission length.

Simulation Results

From Figure 6.4, we can see that as the number of transmission links increases and the network becomes crowded, the number of required time slots for LDS increases much more dramatically than SRA and SEA. This is because as the transmissions are closer together, the chances that $d(T(l_i), T(l_j)) < \mu d(l_i)$ for any two links l_i and l_j increase. The "allowed" subroutine would assign each of these two links a different time slot, even though these two links can be scheduled simultaneously.

6.4 Cluster Chain Topology

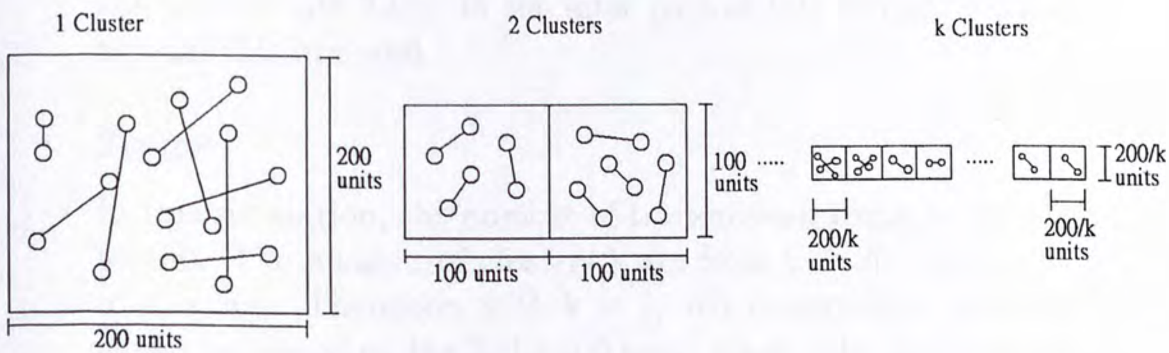

Figure 6.5: In this simulation, requests are spread over to a cluster chain. In this figure, y = 200. Therefore, with k = 1 cluster, n requests would be placed on the 200x200 square plane. In general, $\frac{n}{k}$, $\lceil\frac{n}{k}\rceil$, or $\lfloor\frac{n}{k}\rfloor$ numbers of requests would be placed on $\frac{y}{k}$ by $\frac{y}{k}$ plane, depending on if n is divisible of k.

Purposes

In this simulation, the number of transmission requests is fixed to n. These n requests are taken places into k groups, where each group has $\frac{n}{k}$, $\lceil\frac{n}{k}\rceil$, or $\lfloor\frac{n}{k}\rfloor$, depending on if n is divisible by k. Also, requests in each group would be placed on the $\frac{y}{k}$ by $\frac{y}{k}$ plane, where y is a constant, and forming a cluster. Thus, there are k clusters. These clusters are placing side by side, forming a cluster chain, which is shown in Figure 6.5. For example, if k = 1, then n transmission requests would be randomly placed on one y by y plane. If k = 2, then n/2 requests would be placed on one plane and n/2 requests on another plane, assuming that n is a even number.

It can be seen that as the number of k increases, the transmission requests are spread over a chain. The purpose of the simulation is to investigate the performance of LDS, SRA, and SEA in this situation. In general, as the transmissions are spread over a chain, the number of required time slots to schedule these links would decrease dramatically. However, we will see that this is not the case for LDS. In the later part of this section, we will explain this in detail.

Scopes

In this simulation, the number of transmission requests is fixed to 100. The number of clusters k are from 1 to 20. The value of y is 200. Therefore, with k = 1, 100 transmission requests would be placed on the 200 x 200 $unit^2$ plane. The transmission length of each request is not fixed. All the requests are placed randomly on the clusters.

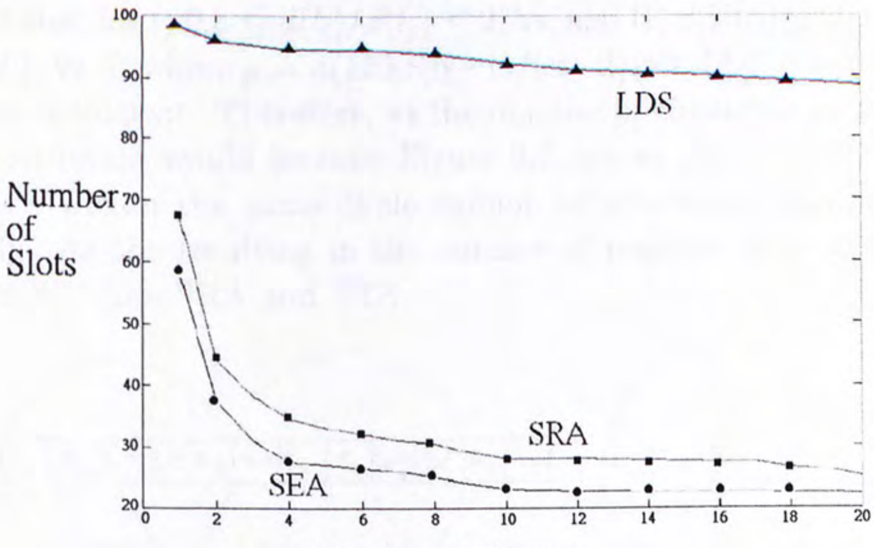

Figure 6.6: Simulation Result using Approximation Method of SEA in general random network.

Simulation Results

Figure 6.6 shows the averaged result of the simulation. It can be seen that there is a sharp decrease in the number of required time slots for SRA and SEA if the number of clusters increases from one to two. This is because the transmission requests are spread over, and therefore, the interference experienced by a receiver decreases, resulting in more simultaneous transmissions. The decreases in the number of required time slots become steady starting from ten clusters and beyond.

However, it can be seen that the decreases in the number of required time slots for LDS are smaller than SRA and SEA. This is because when the number of clusters increases, the plane's area of each cluster decreases. Since each transmission request has to be within this area, the chances of having similar transmission length for requests in a cluster increases. As stated in the Chapter 4 that for I) $0.5 \leq d(l_i)/d(l_j) \leq 2, \forall i$, and II) $d(T(l_i), T(l_j)) < \mu d(l_i), \forall i, j$, where $\mu = 4(120\beta(\alpha-1)/(\alpha-2))^{1/a}$, LDS would become inefficient. Therefore, as the number of clusters increases, the situation would become Figure 6.7, where transmission requests within the same circle cannot be scheduled simultaneously. As the resulting in the number of required time slots a lot more than SRA and SEA.

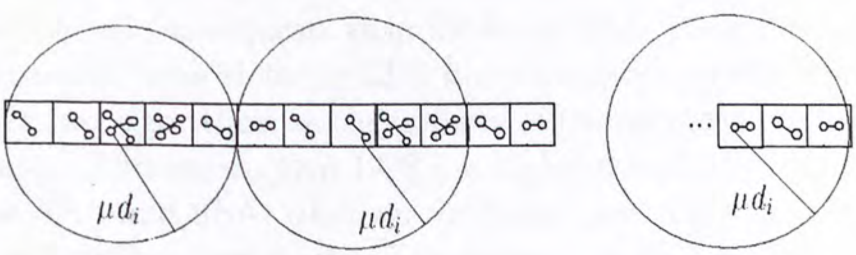

Figure 6.7: Simulation Result using Approximation Method of SEA in general random network.

6.5 General Random Network

Purposes

In this simulation, we would like to investigate the performances of LDS, SRA, SEA in the random network. The purpose of this simulation is to determine the performances of these three algorithms on average.

Scopes

In this simulation, transmission requests are placed randomly on a 200x200 unit plane. Both the x, y coordinates of senders and receivers are generated by MATLAB random function, resulting in the random topology. Coordinates of all devices are unique, meaning that there is no overlapping nodes. For each specific number of transmission requests n, 1000 trials are simulated, and the results are averaged. Figure 6.7 is the averaged results.

Simulation Results

In Figure 6.8, it can be seen that LDS requires more time slots to schedule all the requests than SRA and SEA. Also, the number of required time slots for LDS increases more rapidly than the other two algorithms as the number of transmission requests increases. This means that LDS has higher scheduling complexity than SRA and SEA, while on the other hand, SEA has the lowest scheduling complexity. This shows that SEA has the better performance than SRA and LDS in general.

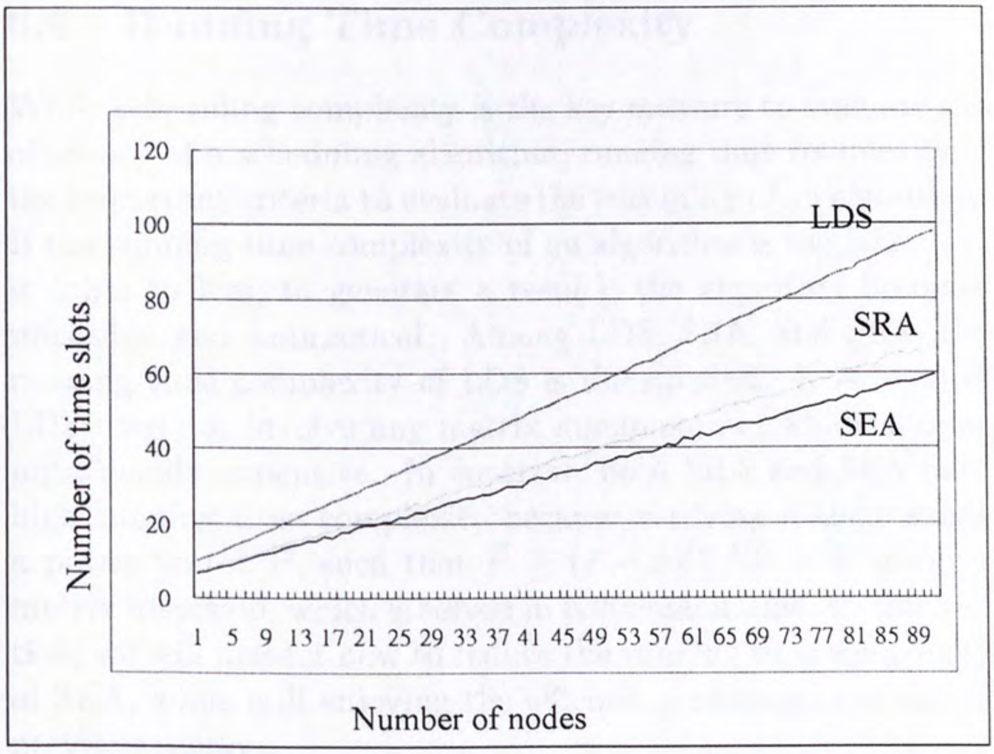

Figure 6.8: Performance evaluation of LDS, SRA, and SEA in general random network topology.

Discussion of Performance and Future Work

When we write our simulation program, we have inserted a boolean statement to check if there exist any network scenarios in which SEA would require more required time slots than LDS amd SRA. If there exist any, the program would return false. However, thus far, we have received no false statement in our simulations, and we would like to have further explorations of this as the future work.

6.6 Running Time Complexity

While scheduling complexity is the key measure to evaluate the efficiency of a scheduling algorithm, running time complexity is the important criteria to evaluate the feasibility of an algorithm. If the running time complexity of an algorithm is too large (i.e. it takes to long to generate a result), the algorithm becomes infeasible and impractical. Among LDS, SRA, and SEA, the running time complexity of LDS is the smallest. It is because LDS does not involve any matrix manipulation, which is computationally expensive. In contrast, both SRA and SEA have high running time complexity because verifying if there exists a power vector $\overline{P}$, such that $\overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0$, involves matrix inversion, which is solved in polynomial time. In this section, we will present how to reduce the running time complexity of SEA, while still enjoying the efficient performance stated in previous sections.

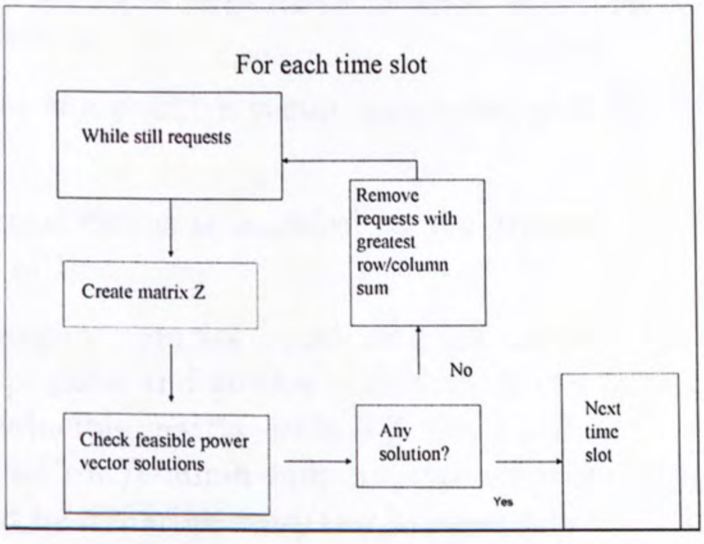

Figure 6.9: Flow chat of SRA in a time slot.

6.5.1 Approximation Algorithm of SRA

Step-Removal Algorithm consists of several key steps, which are shown in Figure 6.8. The most computationally expensive step is to check if there exists a positive power vector in (6.1) for a given Z.

$$\overline{P} \geq (I - \beta Z)^{-1}\overline{N} > 0 \tag{6.1}$$

Since $\bar(N)$ is a positive vector, in order for $\overline{P}$ to be nonnegative, $(I - \beta Z)^{-1}$ has to be positive. Therefore[26]

$$\lambda(\beta Z) < 1 \tag{6.2}$$

. where $\lambda(\beta$ Z) is the eigenvalue of matrix Z.

Approximation algorithm of SRA can be developed based on the Perron-Frobenius Theorem [25] that:

Theorem 1: Perron-Frobenius Theorem states that for irreducible non-negative n x n matrix H,

1. Spectral radius of H, ρ(H), which is largest eigenvalue of H, is positive.
2. There is a positive vector associated with the spectral radius.
3. Spectral radius is bounded by the greatest row or column sum of H.

Except diagonal entries equal zero, all entries of matrix Z in (6.1) are positive and greater than zero. It can been shown that Z is a irreducible matrix, so is β Z. Since $\rho(\beta$ Z) is bounded by the greatest row/column sum, an approximation algorithm can be derived by requiring every row/column sum of β Z less than 1.

$$max\{\sum_i \beta z_{ij}, \sum_j \beta z_{ij}\} < 1 \tag{6.3}$$

Thus, according to the theorem, a positive power vector would be guaranteed for the inequality (6.1).

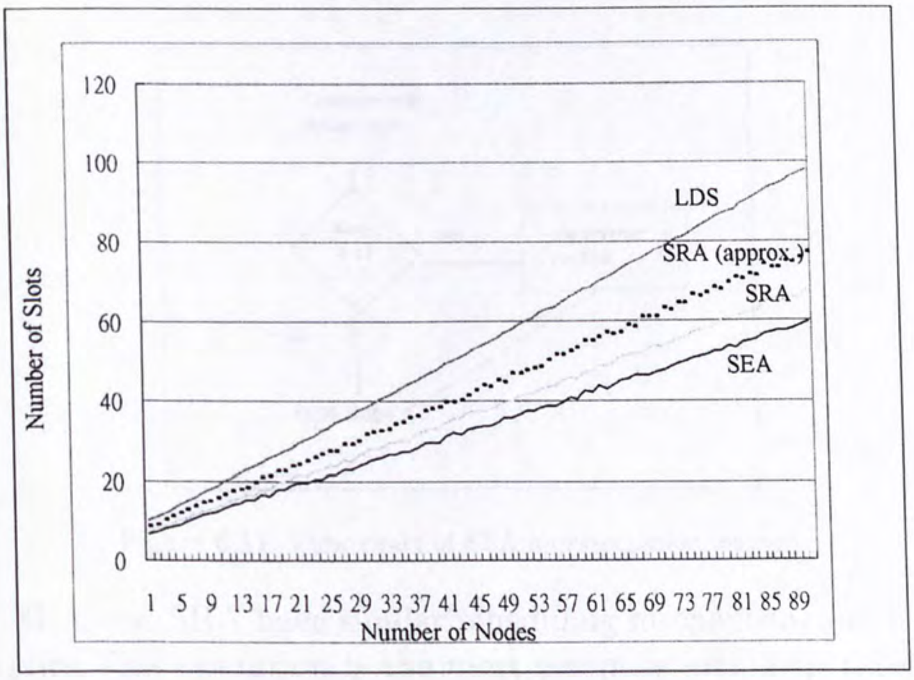

Figure 6.10: Comparison on the SRA approximation method and SRA.

The criteria of (6.2) is only sufficient but not necessary. Therefore, using row/column sum as the criteria to verify if there exists a positive power vector for (6.1) may miss the possible concurrent transmission links. Figure 6.10 shows the simulations of the approximation algorithm of SRA, SRA, SEA, and LDS on the random network for comparison. The simulation has the same assumptions and scopes as section 6.5. It can be seen from the figure that the SRA approximation algorithm requires more time slots to schedule all requests than the original SRA. The difference in the required time slots between SRA approximation and SRA shows the decrease in efficiency when the approximation method is used over the manipulation of matrix inversion.

6.5.2 Approximation Algorithm for SEA

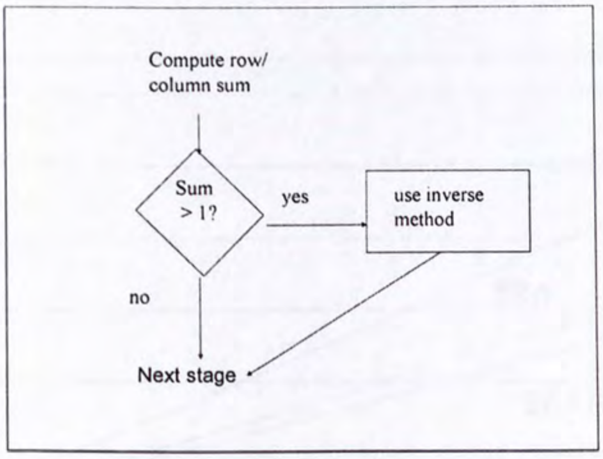

Figure 6.11: Flow chart of SEA approximation method.

SEA and SRA have similar scheduling mechanism, and both require This operation is the most computational expensive in the algorithm. From the previous section, we know that row/column sum method can be used to replace the manipulation of matrix inverse. However, the tradeoff would be the worse resulting performance.

In fact, we can apply the row/column method intelligently to minimize the running time complexity of the verification of (6.1). Figure 6.11 is the flow chart of how it is applied. The idea is that: the row/column method is always applied first, and the matrix-inverse method is applied only if the resulting row/column sum is greater than 1. By doing so, we can reduce the computational cost of power-checking in both SEA and SRA, while at the same time we would not miss any possible solutions. This can be shown from the simulation result of random network in the Figure 6.12. In this simulation, the scope is the same as section 6.5. We can see that the result on Figure 6.12 is approximately the same as Figure 6.5. This means that the approximation algorithm does not worsen the performance of SEA. However, the running time is significantly improved.

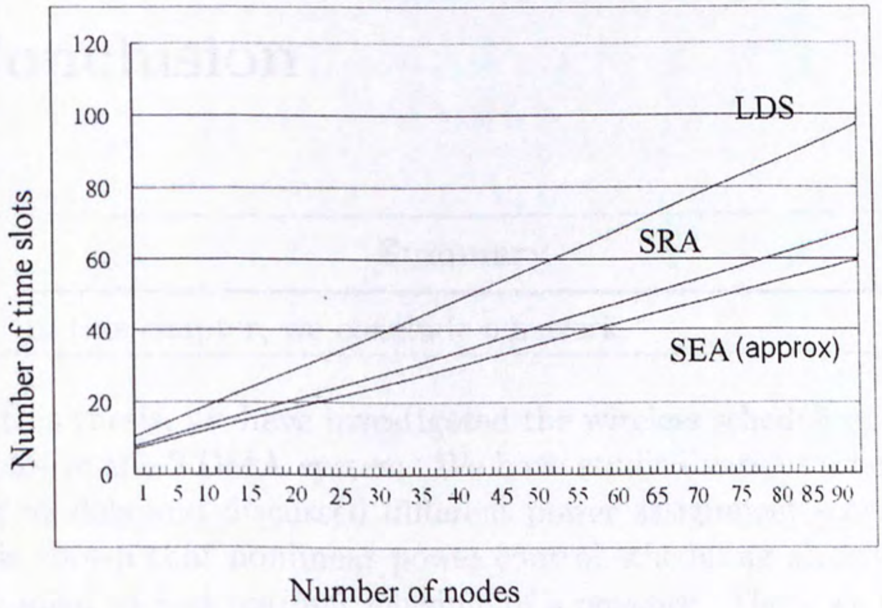

Figure 6.12: Simulation Result using Approximation Method of SEA in general random network.

□ **End of chapter.**

Chapter 7

Conclusion

Summary

In this chapter, we conclude our work.

In this thesis, we have investigated the wireless scheduling protocols in the TDMA system. We have studied various scheduling models and discussed different power assignment schemes. It is shown that nonlinear power control scheduling algorithms are ideal to find optimal schedule of a network. Then, we have discussed two-phase scheduling and step-removal scheduling approaches in nonlinear power assignment scheduling scheme. Also, we have provided analysis of Low-Disturbance Scheduling Algorithm (LDS) and Step-Removal Algorithm (SRA), each of which belongs to the two different scheduling approaches. We have shown that both algorithms performs inefficiently on certain network topologies. It can be summarized as below:

- General two-phase algorithms would over-eliminate potential links which can be scheduled in the same time slot.
- LDS has limitation on networks with transmissions having similar transmission length and senders in close proximity.
- SRA perform efficiently on the exponential chain network.

We have proposed an algorithm called Step-Establishing Algorithm, SEA, which is opposite to SRA and adds links one by one. Although SEA and SRA have similar logic, we have shown that directly applying the grouping logic of SRA would result in inefficient schedule. Thus, we have proposed a new selection condition to group concurrent transmission requests. In the simulations, it is shown that SEA has better performances on the networks which SRA and LDS perform inefficiently, as well as in general random network. Also, in the simulation process, we have found no topology yet SEA performs less efficient than LDS and SRA. As the future work, we would like to investigate if there exists any topology which SEA has worse performance than LDS and SRA.

□ **End of chapter.**

Appendix A

Step-Removal Algorithm

Summary

A.1 Step-Removal Algorithm[1]

Input: set of transmission requests $\wedge$=l
time slot $t := 1$;
while there are links to schedule do
    compute SINR and P from Z;
    while SINR $\leq \beta$ do
        find link l_k, which has the maximal row/column sum in Z
        remove link l_k and corresponding row and col. from Z;
        compute SINR and P from new Z;
    end while
    schedule the links of Z in time slot t and assign P;
    time slot $t := t + 1$;
    compute new Z for unscheduled links;
end while

A.2 Illustration of the efficiency of SRA

To show that SRA is inefficient in the exponential chain topology, we can consider an example with 3 transmissions requests. Consider Figure 2.2 that an exponential chain network with senders in white and receivers in grey. For each link l_i, transmitter $T(l_i)$ and receiver $R(l_i)$ are positioning at -2i and 2i of a chain respectively. (i.e. senders placed at -2^1, -2^2, -2^3 , receivers placing at 2^1, 2^2, 2^3, for transmissions l_1, l_2, l_3, respectively.) We assume there is no power limit in this example, meaning that power can be very large closing to infinite, and assume that the SINR threshold $\beta = 2$, path loss index $\alpha = 4$. Starting with time slot 1, SRA first determines if there is a possible power vector for all four transmission requests, according the inequality of (3.11), which is rewritten below for the convenience of readers.

$$\overline{P} \geq (I - \beta Z)^{-1}\overline{N} \tag{A.1}$$

$$Z = \begin{bmatrix} 0 & 0.3951 & 0.0512 \\ 6.321 & 0 & 0.3951 \\ 13.1072 & 6.3210 & 0 \end{bmatrix} \tag{A.2}$$

Since power has to be positive, with the Z matrix shown in (A.2), it can be determined that there is no possible power vector that satisfies (A.1). Therefore, the SRA would remove the transmission request with the greatest row/column sum, which is request l_3 in this case. From the remaining request l_1 and l_2 , a new Z* is formed.

$$Z^* = \begin{bmatrix} 0 & 0.3951 \\ 6.321 & 0 \end{bmatrix} \tag{A.3}$$

It can be seen that there is also no feasible power vector $\overline{P}$ for l_1 and l_2 such that $\overline{P} \geq (1 - \beta Z)^{-1}\overline{N}$ is satisfied. Therefore, SRA would remove l_2 as it has the greatest row/column sum. Then SRA assigns request l_1 to time slot 1. For the remaining requests l_2 and l_3 with $Z^\sharp$ that

$$Z^\sharp = \begin{bmatrix} 0 & 0.3951 \\ 6.321 & 0 \end{bmatrix} \tag{A.4}$$

It can also been seen that l_2 and l_3 cannot be active concurrently. SRA would assign time slot 2 and 3 to l_2 and l_3 respectively. Therefore, the algorithm takes 3 time slots to schedule all the requests. The result of this example can be extended to general case that SRA takes n time-slot to schedule n transmission requests.

□ **End of chapter.**

Appendix B

Low-Disturbance Scheduling Algorithm

Summary

B.1 Low-Disturbance Scheduling Algorithm

Consider all links l_i in $\wedge$ in decreasing order of $d(l_i)$:
label links such that $d(l_i) \geq d(l_j)$ for $i<j$ and $i \neq j$
$\tau := 1$; $\gamma := 1$; *last* $:= d(l_1)$, the longest transmission length;
for each l_i in $\wedge$ **do**
    if last$/d(l_i) \geq 2$ **then**
        if $\gamma < \lceil log(3n\beta) + \rho log(\alpha) \rceil$ **then**
            $\gamma := \gamma + 1$;
        else
            $\gamma := 1$; $\tau := \tau + 1$;
        end
        last $:= d(l_i)$
    end
γ(i)$:= \gamma$; τ(i)$:= \tau$
end

Main scheduling-loop:

Define constant υ such that $\upsilon := 4N$;
$t := 1$;
for k = 1 to $\lceil log(3n\beta) + \rho log(\alpha) \rceil$ **do**
    Let F_k be the set of all requests l_i with γ(i):= k.
    while not all requests in F_k have been scheduled **do**
        $L_t :=$ null
        Consider all l_i in F_k in decreasing order of d_i:
        if $allowed(l_i, L_t)$ **then**
            $L_t := L_t \cup \{l_i\}$; $F_k := F_k \cup \{l_i\}$;
        end if
        Schedule all $l_i \in E_t$ in time slot t, assigning s_i
        a transmission power of $Pi = \upsilon d_i \alpha (3\upsilon\beta\rho^\alpha)\tau(i)$
        $t := t + 1$;
    end while
end for

allowed(l_i, L_t)

Define constant μ such that $\mu := 4(120\beta(\alpha - 1)/(\alpha - 2))^{1/\alpha}$
for each l_i in F_k **do**
    $\delta_{ij} := \tau(i)$ - $\tau(j)$;
    if $\tau(i) = \tau(j)$ and $\mu d_i > d(T(l_i), T(l_j))$
    or $\tau(i) > \tau(j)$ and $d_i(3n\beta\rho^\alpha)^{(\delta_{ij}+1/)\alpha} > d(T(l_i), R(l_j))$
    or $\tau(i) > \tau(j)$ and $d_j/\rho > d(T(l_i), R(l_j))$
    then return false
end for
return true

Illustration of LDS

To further illustrate how LDS works, consider an example with 10 transmission requests on the exponential chain network, with the assumption that β=2 and α=4 Following the steps of the pre-processing phase, the algorithm first sorts the requests by their transmission distances. Then, starting with the request with the longest transmission length, which is request 10 in this case, the algorithm would group them into γ and τ accordingly. After the pre-processing phase, we would have the γ and τ values as Table B.

Requests with same γ would be passed the "allowed" subroutine and if the subroutine returns true, requests can be transmitted in the same time slot. Those requests that result in false subroutine return would be postponed to next time slot. In this example, link 10 would be assigned to time slot t, then link 6 and link 10 would be passed to "allowed" subroutine. If the subroutine returns true, Link 6 would be assigned time slot t. If it returns false, link 6 would be postponed.

Table B.1: **Notation for the scheduling complexity**($f(n)$=$|S(\wedge)|$, the number of time slots required to schedule a set of transmission requests.)

	γcur	τ cur
Link 1	2	3
Link 2	1	3
Link 3	4	2
Link 4	3	2
Link 5	2	2
Link 6	1	2
Link 7	4	1
Link 8	3	1
Link 9	2	1
Link 10	1	1

□ **End of chapter.**

Bibliography

[1] J. Zander. "Distributed cochannel interference control in cellular radio systems". *IEEE Trans. Veh. Technol.* Vol 41, Aug. 1992.

[2] T.-H. Lee, J. -C. Lin, and Y. T. Su. "Downlink power control algorithms for cellular radio systems". *IEEE Trans, Vech. Technol.*, vol 44, 1995

[3] T. Moscribroda, Y. A. Oswald, R. Wattenhofer. "How optimal are wireless scheduling protocols". *IEEE Conference on Computer Communications (INFOCOM)*, May 2007

[4] J. Gronkvist, A. Hansson. "Comparison between Graph-Based and Interference-Based STDMA scheduling". *ACM MOBIHOC*, 2001.

[5] T. ElBatt., A. Ephremides. "Joint scheduling and power control for wireless ad-hoc networks". *IEEE INFOCOM*, 2001

[6] J. Zander. "Performance of optimum transmitter power control in cellular radio systems". *IEEE Trans. Veh. Technol.*, 41, 1992.

[7] T. Cormen, C. Leiserson, R. Rivest, C. Stein. "Introdiction to algorithm (second edition)". *The MIT press.*2001

[8] A. Behzad and I. Rubin. "Optimum integrated link scheduling and power control for multihop wireless networks" IEEE Trans. Veh. Techol. 56, 2007

[9] T. Moscribroda, R. Wattenhofer. "The complexity of connectivity in wireless networks". *INFOCOMM* 2006

[10] T. ElBatt, A. Ephremides. "Joint scheduling and power control for wireless ad hoc networks".

[11] A. Berman, R.J.Plemmons "Nonnegative Matrices in the Mathematic Sciences", New York: *Academic Press*, 1979.

[12] B. Korte, J. Vygen "Combinatorial Optimization: Theory and Algorithms", *Springer*, 2002

[13] H. Zhang. Service disciplines for guaranteed performance service in packet switching networks. *Proceedings of the IEEE*, October 1995

[14] M. Gerharz, C. Waal, M. Frank, P. James. A practical view on Quality-of-Service Support in Wireless Ad Hoc Networks. *Proc. Of the 3rd IEEE Workshop on Applications and Services in Wireless Networks.* P.185-196, July 2003

[15] T. S. Rappaport. Wireless Communications, Principles and Practices, second edition. *Pearson Education*, 2004.

[16] Y. Okumura et. Al. "Field strength and its variability in UHF and WHT land-mobile radio service." *Rev. Elec. Commun. Lab*, vol. 16, 1968

[17] T.K. Sarkar, et al., "A survey of various propagation models for mobile communication," *Antennas and Propagation Magazine*, vol. 45, Issue 3, June 2003, pp. 51-82

[18] S. Singh and C. S. Raghavendra. PAMAS - Power Aware Multi- Access Protocol with Signalling for Ad Hoc Networks. *SIGCOMM Comput.Commun. Rev.*, 28(3):5-26, 1998.

[19] A. Woo and D.-E. Culler. A Transmission Control Scheme for Media Access in Sensor Networks. *In Proceedings of the*

7th International Conference on Mobile Computing and Networking (MOBICOM), pages 221-235, 2001.

[20] P. Monks, V. Bharghavan, and W. W. Hwu. A Power Controlled Multiple Access Protocol for Wireless Packet Networks. *In Proceedings of the 20th Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pages 1567-1576, 2001.

[21] S. Banerjee and A. Misra. Minimum Energy Paths for Reliable Communication in Multi-Hop Wireless Networks. *In Proceedings of the 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, pages 146-156, 2002.

[22] W. Ye, J. Heidemann, and D. Estrin. An Energy-Efficient MAC Protocol for Wireless Sensor Networks. *In Proceedings of the 21th Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, pages 1567-1576, 2002.

[23] A. Behzad and I. Rubin. On the Performance of Graph-based Scheduling Algorithms for Packet Radio Networks. *In Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, pages 3432-3436, 2003.

[24] M. T. Hajiaghayi, N. Immorlica, V. S. Mirrokni. Power Optimization in Fault-Tolerant Topology Control Algorithms for Wireless Multi-hop Networks. *Mobicom 03*, 2003

[25] S. U. Pillai, T. Suel, and S. Cha. The Perron-Frobenius Theorem, Some of Its Application . *IEEE Signal Processing Magazine*, pages 62, March 2005.

[26] T. Holliday, N. Bambos, P. Glynn, and A. Goldsmith. Distributed Power Control for Time Varying Wireless Net-

works: Optimality and Convergence. *Proc. Allerton Conf. Commun., Control, and Computing*, 2005

[27] C.K. Toh. Ad Hoc Mobile Wireless Networks. *Prentice Hall Publishers*, 2002

[28] M. Ghaderi and R. Boutaba. Call Admission Control in Mobile Cellular Networks: A Comprehensive Survey. *In Press. International Journal on Wireless Communications and Mobile Computing. John Wiley*, 2004.

[29] L. F. Akyildiz, X. Wang, W. Wang. Wireless Mesh Networks: A survey. *Computer Networks,* 47:445–487, 2005

[30] I. Katzela, M.Naghshineh. Channel assignment schemes for cellular mobile telecommunicationsystems: a comprehensive survey. *Personal Communications, IEEE* vol. 3, Issue 3, Jun 1996

[31] J. Grönkvist. Distributed Scheduling For Mobile Ad Hoc Networks – A Novel Approach. *Personal, Indoor and Mobile Radio Communications*, 2004.